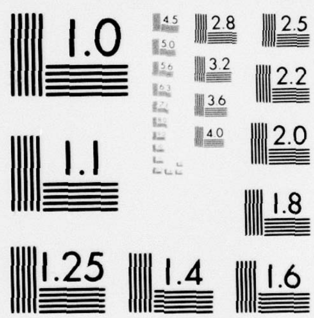
AD-A077 148 ILLINOIS UNIV AT URBANA-CHAMPAIGN DECISION AND CONTROL LAB F/G 13/8
SENSITIVITY ADAPTIVE CONTROL OF A MAGNETIC SUSPENSION SYSTEM.(U)
MAY 79 Y M CHAN DAAG29-78-C-0016

UNCLASSIFIED DC-25 NL

1 OF 2
AD
A077148



MICROCOPY RESOLUTION TEST CHART

REPORT DC-25

CSL COORDINATED SCIENCE LABO

DECISION AND CONTROL LABORAT

LEVEL II

SENSITIVITY ADAPTIVE CONTROL OF A MAGNETIC SUSPENSION SYSTEM

YICK MAN CHAN

REPORT R-843

UILU-EN

UNIVERSITY OF ILLINOIS – URBANA,

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)

SENSITIVITY ADAPTIVE CONTROL OF A MAGNETIC SUSPENSION SYSTEM

5. TYPE OF REPORT & PERIOD COVERED

Technical Report

6. PERFORMING ORG. REPORT NUMBER

R-843(DC-25); UILU-78-2236

7. AUTHOR(s)

YICK MAN CHAN

8. CONTRACT OR GRANT NUMBER(s)

AFOSR-78-3633
DAAG-29-78-C-0016
NSF-ENG-74-20091
NSF INT-77-20969

9. PERFORMING ORGANIZATION NAME AND ADDRESS

Coordinated Science Laboratory
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS

Joint Services Electronics Program

12. REPORT DATE

May 1979

13. NUMBER OF PAGES

89

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)

15. SECURITY CLASS. (of this report)

UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

Master's thesis,

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

DC-25, UILU-ENG-78-2236

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Stochastic Adaptive Control
Sensitivity Adaptive Feedback
Kalman Filtering
Factorization Method
Control of Systems with Unknown Parameters

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This project is concerned with the study of the behavior of a particular stochastic system, the magnetic suspension system, under the application of the Sensitivity Adaptive Feedback with Estimation Redistribution (SAFER) control algorithm. Matrix factorization techniques are used in the controller and estimator design for the system. Simulation results indicate that the magnetic suspension system can operate satisfactorily under the SAFER control method; and factorization techniques indeed enhance the numerical stability of computations. However, due to the complex structure of the SAFER control

UILU-ENG 78-2236

SENSITIVITY ADAPTIVE CONTROL OF A MAGNETIC SUSPENSION SYSTEM

by

Yick Man Chan

This work was supported in part by the Air Force Office of Scientific Research under Contract AFOSR-78-3633; the Joint Services Electronics Program (U.S. Army, U.S. Navy and U.S. Air Force) under Contract DAAG-29-78-C-0016; the National Science Foundation under Grants NSF-ENG-74-20091 and NSF INT-77-20969.

SENSITIVITY ADAPTIVE CONTROL

OF A MAGNETIC SUSPENSION SYSTEM

BY

YICK MAN CHAN

B.S., Rose-Hulman Inst. of Tech., 1977

THESIS

Submitted in partial fulfillment of the requirements for the degree of Master of Science in Electrical Engineering in the Graduate College of the University of Illinois at Urbana-Champaign, 1979

Thesis Advisor: Professor J. B. Cruz, Jr.

Urbana, Illinois

ABSTRACT

This project is concerned with the study of the behavior of a particular stochastic system, the magnetic suspension system, under the application of the Sensitivity Adaptive Feedback with Estimation Redistribution (SAFER) Control Algorithm. Matrix factorization techniques are used in the controller and estimator design for the system. Simulation results indicate that the magnetic suspension system can operate satisfactorily under the SAFER control method; and factorization techniques indeed enhance the numerical stability of computations. However, due to the complex structure of the SAFER control method, real time application of the algorithm may require faster computing device or simplified mathematical model. Its application will be most pertinent to system with slow time constants.

ACKNOWLEDGEMENT

The author would like to express his appreciation towards his advisor, Professor J. B. Cruz, Jr. for his patience and most beneficial guidance and advice throughout the thesis work. Finally, the author would like to thank Wilma McNeal for a very fine typing job.

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

CHAPTER 1

INTRODUCTION

With the advent of sophisticated digital computers, it is now possible to implement more complex adaptive controllers. Adaptive controllers are appealing as they are able to modify their behavior depending on the performance of the control led process, thus eliminating or reducing the need for manual tuning of the parameters of the controllers. One reason for the presence of these controllers in the process industry is that the dynamics of the process or the noise are changing due to changes in production or wear of the equipment. The use of adaptive controllers takes these changes into account and arrive at an optimal control according to a prescribed manner.

1.1. Adaptive Controllers

Many adaptive controllers are constructed by taking into consideration the statistical nature of the fluctuation of the parameters or the disturbances acting on the system; this class of controllers is called stochastic adaptive controllers. Stochastic adaptive controllers can further be divided into non-dual and dual controllers according to the structure of the performance index imposed on the system under consideration. If the performance index only takes into account the previous measurements and does not assume that further information will be available, then the resulting controller are called non-dual. On the other hand, the performance index can also be dependent on the future observations and this will result in a dual controller. Hence, the minimization of a loss function (performance index), when the measurement program is incorporated, several steps ahead will give a dual controller and it might be worthwhile for the controller to take some actions in order to improve the estimates of the fluctuating parameters, which may also be unknown. That is, the dual controller must ensure good control and good estimation, which in general are contradictory since good estimation might require large control signals while good control might require that the control signals to be small. A dual controller thus must compromise between these two tasks [16].

1.2. Adaptive Sensitivity Controller

One method of constructing a dual controller for an uncertain system with unknown parameters is to design a control algorithm based on the information extracted from the sensitivities of the system [12,13,14]. This method, the Sensitivity Adaptive Feedback with Estimation Redistribution (SAFER) control algorithm, incorporates a cost assignment for the estimation effort and the estimation budget is distributed according to the accuracy required to achieve a given control objective. The rationale underlying this method is that parameters to which the state of the system is more sensitive require more accurate estimation than those whose effect on the state is less significant. The fixed budget for estimation is represented by a sensitivity constraint. A suboptimal scheme is then used to compute a control (which will in turn influence the sensitivity of the system) that will minimize the covariance of the estimate of the unknown parameters.

1.3. Thesis Outline

In this thesis, a stochastic model for a magnetic ball suspension system is studied and simulated using a digital computer. The control algorithm used is the previously mentioned SAFER method. The objective of the project is to investigate the performance of the magnetic suspension system under the application of the SAFER control algorithm. In addition, matrix factorization techniques are used in some parts of the controller and estimator design to determine if numerical stability can be enhanced.

The actual non-linear model of the magnetic ball suspension system is first linearized, then discretized. A set of feedback gains is then computed using the SAFER method. Finally, an extended Kalman filter utilizing matrix factorization algorithm is constructed for estimating the unknown parameters of the model. Monte Carlo simulations are then carried out with this closed-loop system. Various results, comments and remarks on the system or the method are mentioned in subsequent chapters of the thesis.

CHAPTER 2

MAGNETIC BALL SUSPENSION MODEL AND PROBLEM FORMULATION

The magnetic ball suspension system is studied since in many situations it is desirable to support an object with a magnetic field; for example, in a wind tunnel, effects of the mechanical structure (stinger) supporting the model under study introduce errors in drag and lift measurements. One solution is naturally to use a magnetic field to support the model in position [5]. Another application of this system is that it serves as a simplified model for a magnetically suspended vehicle [9].

2.1. Equation of Motion

The equation of motion for the magnetic ball suspension system has the form

$$m \frac{d^2h}{dt^2} = - \frac{ci^2}{h} + mg \tag{2.1}$$

where m is the mass of the ball, g is the gravitational constant, h is the distance of the ball from the coil, i is the current in the coil which generates the electromagnetic force and c is a proportionality constant which is dependent upon the particular configuration and construction of the coil, number of turns per centimeter, for instance. The derivative in the equation is taken with respect to time, thus, the second derivative of h with respect to time t will be the acceleration of the ball. An illustration of the configuration is shown in Figure 1.

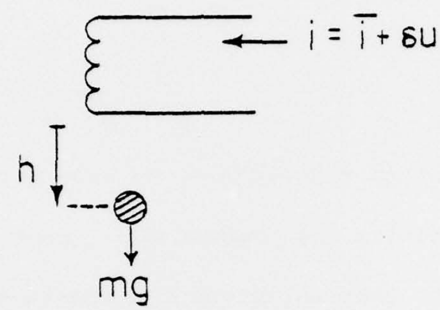


Fig. 1. Magnetic Ball Suspension System Configuration

To formulate the problem using the state space approach, let

$$x_1 = h$$

$$x_2 = \frac{dh}{dt} = \dot{x}_1$$

$$u = i$$

then equation (2.1) can be transformed into the following second order system:

$$\dot{x}_1 = x_2 \tag{2.2}$$

$$\dot{x}_2 = -\frac{ci^2}{mx_1} + g \tag{2.3}$$

2.2. Linearization

Since the control algorithm to be used applies to a linear system, it is necessary to linearize the present non-linear system using first order perturbation techniques, that is, given the non-linear system

$$\dot{x} = f(x,u) \tag{2.4}$$

we linearize it around an equilibrium position and obtain

$$\delta\dot{x} = \frac{\partial f}{\partial x}\,\delta x + \frac{\partial f}{\partial u}\,\delta u = C\,\delta x + D\,\delta u \tag{2.5}$$

where

$$C = \frac{\partial f}{\partial x}, \quad D = \frac{\partial f}{\partial u}.$$

Before computing the various partial derivatives, an equilibrium state of the system has to be evaluated. This is done by setting, in the original non-linear system, $\dot{\bar{x}} = f(\bar{x},\bar{u}) = 0$, where $\bar{x}$ and $\bar{u}$ are equilibrium values.

Setting equation (2.2) and (2.3) equal to zero, we obtain the equilibrium values for x_1 and x_2:

$$\bar{x}_2 = 0 \tag{2.6}$$

$$\bar{x}_1 = \frac{c\bar{i}^2}{mg} \tag{2.7}$$

where $\bar{i}$ is the bias current in the coil.

Computing the various partial derivatives, the following linearized system is obtained:

$$\delta\dot{x}_1 = \delta x_2 \tag{2.8}$$

$$\delta\dot{x}_2 = \frac{c\bar{i}^2}{m\bar{x}_1^2}\,\delta x_1 - \frac{2c\bar{i}}{m\bar{x}_1}\,\delta u \tag{2.9}$$

Substituting equation (2.7) into (2.9), we obtain

$$\delta\dot{x}_2 = \frac{mg^2}{c\bar{i}^2}\,\delta x_1 - \frac{2g}{\bar{i}}\,\delta u \tag{2.10}$$

Following the notation in equation (2.5), we have

$$C = \begin{bmatrix} 0 & 1 \\ \frac{mg^2}{c\bar{i}^2} & 0 \end{bmatrix} \qquad D = \begin{bmatrix} 0 \\ \frac{-2g}{\bar{i}} \end{bmatrix}$$

To simplify the notation, we will denote hereafter, without ambiguity, δx simply by x, δu by u, that is, equation (2.10) will appear in the form $\dot{x} = Cx + Du$.

To attain a more realistic model for the system, we will take into consideration the effect of disturbances, such as drag force and transducer measurement error, by appending a noise term w into equation (2.10), which is the linearized version of the original system. Hence, the following linearized system with noise is obtained

$$\dot{x} = Cx + Du + Fw \tag{2.11}$$

where

$$F = \begin{bmatrix} 0 \\ 1 \end{bmatrix} .$$

2.3. Discretization

As the control algorithm to be applied deals with discrete system, it is necessary to discretize the present linearized system (2.11). In order to simplify computations, discretization by the method of finite difference is used. That is, we will approximate $\dot{x}$ by $(x_{k+1} - x_k)/T$, where T is the sampling period. Applying this technique to equation (2.11), we obtain

$$x_{k+1} = (I + CT)x_k + DTu_k + FTw_k$$

or

$$x_{k+1} = Ax_k + Bu_k + Ew_k \tag{2.12}$$

where I is the identity matrix and $B = \begin{bmatrix} 0 \\ \frac{-2gT}{\bar{i}} \end{bmatrix} = \begin{bmatrix} 0 \\ \theta_2 \end{bmatrix}$

$$A = \begin{bmatrix} 1 & T \\ \frac{mg^2 T}{c\bar{i}^2} & 1 \end{bmatrix} = \begin{bmatrix} 1 & T \\ \theta_1 & 1 \end{bmatrix}$$

The two constants θ_1 and θ_2 are the unknown parameters of the discrete stochastic system of equation (2.12). The numerical values of the various variables in the model are listed in Table 1. This discrete stochastic system can be modelled as illustrated in the block diagram given in Figure 2.

Table 1

Numerical values of the variables in the model and the corresponding values of the unknown parameters

$m = 0.05$ Kg.

$g = 10.$ meters/second^2

$c = 0.4$ Newton-meter/ampere^2

$\bar{i} = 0.5$ ampere

$T = 0.01$ second

$\bar{x}_1 = 0.2$ meter

$\theta_1 = 0.5$ second/ampere^2

$\theta_2 = -0.4$ meter/second-ampere

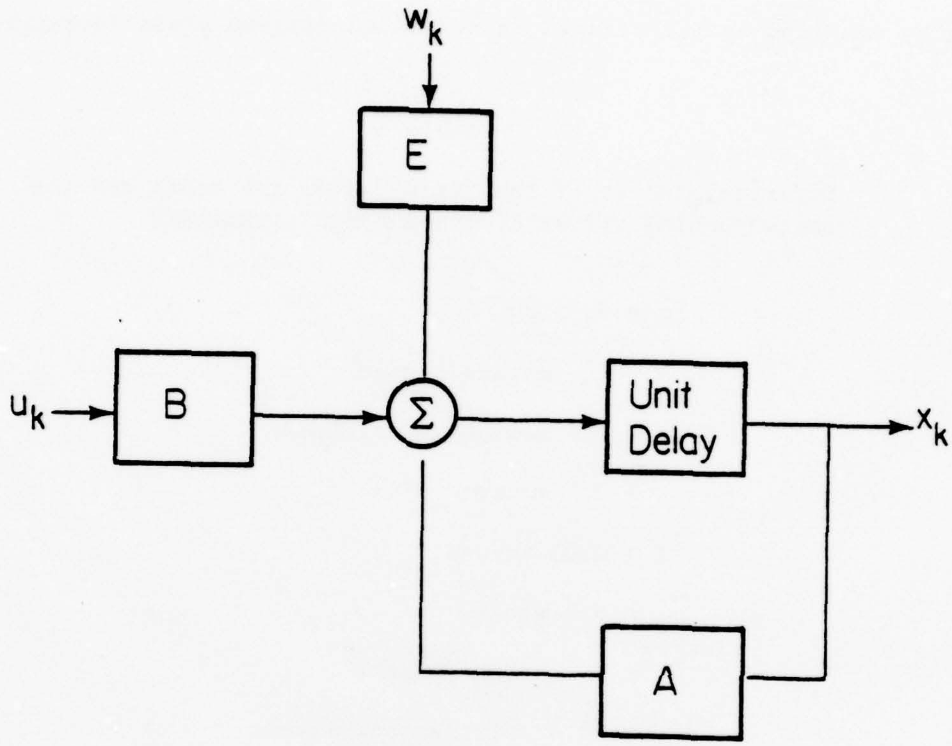


Fig. 2. Discrete Stochastic Plant Block Diagram

CHAPTER 3

CONTROLLER DESIGN

3.1. Sensitivity Feedback Controller

The use of sensitivity functions in control systems to attain a more stable design has been well investigated [6]. In this thesis, the control algorithm to be applied will utilize the sensitivity functions in such a way that a fixed budget of effort for estimation of unknown parameters will be distributed rationally to certain states of the system based on information extracted from the sensitivity functions. In this chapter we apply the SAFER control method in references [12,13,14] using the notation found therein. We use the U-D factorization method [2,3] to improve the numerical stability of the computations.

3.1.1. Problem Formulation and Solution

Given the stochastic model

$$x_{k+1} = Ax_k + Bu_k + Ew_k$$

where A, B, E are defined previously in equation (2.12) and an observation equation

$$c_k = x_k \tag{3.1}$$

The model considered here is a second order system with scalar control and scalar noise. The noise w_k is assumed to be a random sequence with zero mean and variance V_w. The state and the disturbance at the same instant are statistically independent. The entries of A and B are considered independent. The performance index to be minimized is:

$$J_1 = E\{\sum_{k=N_o}^{N_o+\nu-1} x_k'Qx_k + u_k'RU_k\}$$

subject to

$$J_2 = E\{\sum_{k=N_o}^{N_o+\nu-1} \rho'(k)W^{N_o}\rho(k)\} \leq r$$

where $E\{.\}$ denotes taking expectation, ν is the number of stages in the optimization procedure, and

$$\rho(k) = (\rho_1'(k),\rho_2'(k))' = \left\{ \begin{bmatrix} \frac{\partial x_1(k)}{\partial\theta_1} \\ \frac{\partial x_2(k)}{\partial\theta_1} \end{bmatrix}, \begin{bmatrix} \frac{\partial x_1(k)}{\partial\theta_2} \\ \frac{\partial x_2(k)}{\partial\theta_2} \end{bmatrix} \right\},$$

where ρ_1, ρ_2 are designed to closely approximate the state sensitivities by having them satisfy the following equations:

$$\rho_1(k+1) = (A_{\theta_1} - B_{\theta_1}K_1(k))x_k + [\hat{A} - \hat{B}K_1(k)]\rho_1(k) - \sum_{i=1}^{2} B_{\theta_1}K_{2i}(k)\rho_i(k) \tag{3.2}$$

$$\rho_2(k+1) = (A_{\theta_2} - B_{\theta_2}K_1(k))x_k + [\hat{A} - \hat{B}K_1(k)]\rho_2(k) - \sum_{i=1}^{2} B_{\theta_2}K_{2i}(k)\rho_i(k) \tag{3.3}$$

where A_{θ_j} and $B_{\theta_j}(j=1,2)$ are the derivatives of A and B with respect to θ_j, that is:

$$A_{\theta_1} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \qquad B_{\theta_1} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{3.4}$$

$$A_{\theta_2} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \qquad B_{\theta_2} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{3.5}$$

and $\hat{A}$, $\hat{B}$ are the latest estimates of A and B respectively. θ_j is a component of the vector θ which is formed by the unknown entries of A and B taken by row, that is, for the magnetic ball suspension system:

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} mg^2T/(c\bar{i}^2) \\ -2gT/\bar{i} \end{bmatrix} = \begin{bmatrix} 0.5 \\ -0.4 \end{bmatrix} \tag{3.6}$$

The matrices $K_1(k)$ and $K_2(k)$ in equation (3.2) and (3.3) are to be found from the sensitivity control algorithm, and the control signal will be in the form

$$u_k = -K_1(k)x_k - \sum_{i=1}^{2} K_{2i}(k)\rho_i(k) \tag{3.7}$$

The matrix Q is positive semidefinite while R is positive definite. The unknown diagonal matrix W^{N_o} is to be chosen by the designer at time N_o and satisfies $W_i^{N_o} \geq \mathcal{E} \geq 0$ for $i = 1,2,3,4$, and $\text{trace}\{W^{N_o}\} = 1$. The non-negative number $\mathcal{E}$ is the minimum relative weight. It is the choice of this weighting matrix W^{N_o} that enables the distribution of the estimation effort to be done rationally.

Furnished with all the previous information, the original problem can be restated as follows:

Consider the augmented system

$$\xi_{k+1} = \tilde{A}_k \xi_k + \tilde{B}_k u_k + \Gamma w_k \tag{3.8}$$

with perfect state information. The performance index to be minimized with respect to u_k and W^{N_o} is:

$$J = E\{\sum_{k=N_o}^{N_o+\nu-1} (\xi_k' \tilde{Q}_k \xi_k + u_k' R u_k)\} \tag{3.9}$$

where

$$\tilde{A}_k = \begin{bmatrix} A & 0 & 0 \\ A_{\theta_1} - B_{\theta_1} K_1(k) & \hat{A} - \hat{B}K_1(k) - B_{\theta_1} K_{21}(k) & -B_{\theta_1} K_{22}(k) \\ A_{\theta_2} - B_{\theta_2} K_1(k) & -B_{\theta_2} K_{21}(k) & \hat{A} - \hat{B}K_1(k) - B_{\theta_2} K_{22}(k) \end{bmatrix}$$

$$\tilde{B}_k = \begin{bmatrix} B \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \Gamma = \begin{bmatrix} E \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \tilde{Q}_k = \begin{bmatrix} Q & 0 \\ 0 & \lambda W^{No} \end{bmatrix} \tag{3.10}$$

3.1.2. Suboptimal Solution

Since the exact optimal solution of equation (3.9) subject to equation (3.8) is extremely involved, a suboptimal solution which is of the feedback form will be sought instead. Before going into the procedure for finding this suboptimal solution, a theorem found in references [12,13] will be stated since it will be used during the solution process. The theorem shows that for a fixed W^{N_o} and given $\hat{A}_{N_o}$ and $\hat{B}_{N_o}$, if equation (3.13) admits a solution, the control u_{N_o} that minimizes J in equation (3.9) is

$$u_{N_o} = -(R + \tilde{B}'^{N_o}_{N_o} P^{N_o}_{N_o+1} \tilde{B}^{N_o}_{N_o})^{-1} (\tilde{A}'^{N_o}_{N_o} P^{N_o}_{N_o+1} \tilde{B}^{N_o}_{N_o})' \xi_{N_o} \tag{3.11}$$

where $P_i^{N_o}$ is the solution to

$$\begin{aligned} P_i^{N_o} &= \tilde{Q}_i + \tilde{A}_i'^{N_o}(K_i) P_{i+1}^{N_o} \tilde{A}_i^{N_o}(K_i) \\ &\quad - (\tilde{A}_i'^{N_o}(K_i) P_{i+1}^{N_o} \tilde{B}_i^{N_o})' (R + \tilde{B}_i'^{N_o} P_{i+1}^{N_o} \tilde{B}_i^{N_o})^{-1} (\tilde{A}_i'^{N_o}(K_i) P_{i+1}^{N_o} \tilde{B}_i^{N_o})' \\ P^{N_o}_{N_o+\nu-1} &= \tilde{Q}_{N_o+\nu-1} \quad i = N_o, \ldots, N_o + \nu - 2 \end{aligned} \tag{3.12}$$

and

$$\begin{aligned} K_i &= (R + \tilde{B}_i'^{N_o} P_{i+1}^{N_o} \tilde{B}_i^{N_o})^{-1} (\tilde{A}_i'^{N_o}(K_i) P_{i+1}^{N_o} \tilde{B}_i^{N_o})' \\ K_{N_o+\nu-1} &= 0 \quad i = N_o, \ldots, N_o + \nu - 2 \end{aligned} \tag{3.13}$$

where

$$K_i = [K_1(i) \quad K_{21}(i) \quad K_{22}(i)]$$

$$\tilde{B}_k^{N_o} = \begin{bmatrix} \hat{B}_{N_o} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\tilde{A}_k^{N_o} = \begin{bmatrix} \hat{A}_{N_o} & 0 & 0 \\ A_{\theta_1} - B_{\theta_1}K_1(k) & \hat{A}_{N_o} - \hat{B}_{N_o}K_1(k) - BK_{21}(k) & -B_{\theta_1}K_{22}(k) \\ A_{\theta_2} - B_{\theta_2}K_1(k) & -B_{\theta_2}K_{21}(k) & \hat{A}_{N_o} - \hat{B}_{N_o}K_1(k) - B_{\theta_2}K_{22}(k) \end{bmatrix}$$

$\hat{A}_{N_o}$ and $\hat{B}_{N_o}$ are the estimates of A and B given the information up to N_c. It can be easily discerned that the matrix $\tilde{A}_k^{N_o}$ closely resembles to $\tilde{A}_k$ in equation (3.8) with the difference that the estimates for A and B are used instead of the exact matrices. This is, indeed, one reason for the solution to be suboptimal.

The above theorem provides a method to determine a control for a fixed W^{N_o}. However, before the outline for the entire suboptimal is described, it will be necessary to compute the cost (performance index) associated with the specific choice of W^{N_o}. Naturally, the specific W^{N_o} which results in the minimum cost will be chosen as the optimum W^{N_o}, and the control which is associated with this W^{N_o} will be the optimal control signal at stage N_o. The derivation of the equations for evaluating the cost can be found in Appendix A. This cost is

$$J_{N_o} = \xi'_{N_o}\bar{P}_{N_o}\xi_{N_o} + V_w \,\text{trace}\{\Gamma'(\sum_{i=N_o+1}^{N_o+\nu-1}\bar{P}_i)\Gamma\} \tag{3.14}$$

where $\bar{P}_i$ satisfies the following propagation equation

$$\bar{P}_i = \tilde{Q}_i + K_i'RK_i + (\tilde{A}_i^{N_o}(K_i)-\tilde{B}_i^{N_o}K_i)'\bar{P}_{i+1}(\tilde{A}_i^{N_o}(K_i)-\tilde{B}_i^{N_o}K_i)$$

$$\bar{P}_{N_o+\nu-1} = \tilde{Q}_{N_o+\nu-1} \qquad i = N_o,\ N_o+1,\ldots,N_o+\nu-2 \tag{3.15}$$

Equipped with the previously stated theorem and equations for computing the performance index, a procedure for finding a suboptimal solution for minimizing J in equation (3.9) subject to equation (3.8) is outlined as follows:

a) Compute the estimates $\hat{A}_{N_o}$ and $\hat{B}_{N_o}$ based on the information received up to N_o.

b) Choose a specific W^{N_o}.

c) Apply the theorem, that is, use equations (3.11), (3.12), (3.13) to solve for the sequence K_k, $k = N_o+\nu,\ldots,N_o$ backward in time, and compute u_{N_o}.

d) Find the corresponding cost J_{N_o} from equations (3.14) and (3.15).

e) Repeat steps (b) through (d) for the other finite number of choices of W^{N_o}.

f) Apply the control u_{N_o}, which correspond to the value of W^{N_o} which results in the minimum cost J_{N_o}, at time N_o to the system. This yields the optimum W^{N_o} and u_{N_o}.

3.1.3. Feedback Solution to the Magnetic Ball System

In the magnetic ball suspension model, since there are two states and two unknown parameters, there are four sensitivity states and an augmented system of sixth order. Hence, the feedback gain K_i is a row matrix with six entries. With the matrix $P_i^{N_o}$ in equation (3.12) denoted by $P_i^{N_o} = [p_{jl}(i)]$, $\tilde{B}_i^{N_o} = [0,b_2(i),0,0,0,0]'$ and

$$\hat{A}_{No} = \begin{bmatrix} a_1(i) & a_2(i) \\ a_3(i) & a_4(i) \end{bmatrix}$$

where $a_3(i) = \theta_1$, $b_2(i) = \theta_2$ are the unknown parameters, and also with the feedback gains $K_i = [K_1(i) \quad K_{21}(i) \quad K_{22}(i)]$ denoted by $K_i = [k_1(i)\ k_2(i)\ k_3(i)\ k_4(i)\ k_5(i)\ k_6(i)]$, equation (3.13) can be solved sequentially as:

$$k_1(i) = \frac{a_1(i)b_2(i)p_{12}(i+1) + a_3(i)b_2(i)p_{22}(i+1)+b_2(i)p_{42}(i+1)}{(s)(t)}$$

$$k_2(i) = \frac{a_2(i)b_2(i)p_{12}(i+1) + a_4(i)b_2(i)p_{22}(i+1)}{(s)(t)}$$

$$k_3(i) = \frac{a_1(i)b_2(i)p_{32}(i+1) + (a_3(i)-b_2(i)k_1(i))(b_2(i)p_{42}(i+1))}{(s)(t)}$$

$$k_4(i) = \frac{a_2(i)b_2(i)p_{32}(i+1) + (a_4(i)-b_2(i)k_2(i))(b_2(i)p_{42}(i+1))}{(s)(t)}$$

$$k_5(i) = \frac{a_1(i)b_2(i)p_{52}(i+1) + (a_3(i)-b_2(i)k_1(i)(b_2(i)p_{62}(i+1))}{(s)(t)}$$

$$k_6(i) = \frac{a_2(i)b_2(i)p_{52}(i+1) + (a_4(i)-b_2(i)k_2(i))(b_2(i)p_{62}(i+1))}{(s)(t)} \tag{3.16}$$

where

$$S = 1 + \frac{b_2(i)p_{62}(i+1)}{R + P_{22}(i+1)b_2^2(i)}$$

$$t = R + p_{22}(i+1)b_2^2(i)$$

for $i = N_o, \ldots N_o + \nu - 2$, $K^{N_o}_{N_o + \nu - 1} = 0$.

Hence, given a fixed W^{N_o} and the terminal condition in equation (3.12), that is, $P^{N_o}_{No+\nu-1} = \tilde{Q}_{No+\nu-1} = \tilde{Q}_i = \tilde{Q}$, where $\tilde{Q}$ is constant for the system under consideration, equation (3.13) can be solved by cycling through the sets of equations in (3.16) sequentially. The resulting feedback gains are then subsitituted into the matrix $\tilde{A}_i^{N_o}(K_i)$ in equation (3.12) and $P_i^{N_o}$ is then obtained. Cycling through equations (3.12) and (3.13) will result in K_{N_o} which is then subsitituted into equation (3.11) to obtain the control u_{N_o}. A list of the numerical values of the matrices involved in equations (3.12) and (3.13) are tabulated in Appendix B. The various choices (corners) of the weighting matrix W^{N_o} used in the simulation work are also listed in Appendix B.

As the model studied in this thesis consists of only a scalar control, it is readily observed that the matrix inversion in equation (3.12) reduces to a simple scalar division. In the general case where multi-variable control is involved, one can cycle through equations (3.12) and (3.13) by considering only one column of the $\tilde{B}_i^{N_o}$ matrix during each cycle. By going through equations (3.12) and (3.13) m times, where m is the number of control variables, the m control values can be computed involving only scalar division, thus avoiding matrix inversion.

3.2. Factorization Method

During recent years, it has come to be realized that algorithm formulations that make use of matrix factorization generally enhance numerical stability [2,3]. Hence, in going through the computations in equation (3.12) for the covariance update, covariance factorization technique [3,8] and the rank-one update factorization technique [2] is used. The method is described as follows:

1. Assume $P_{i+1}^{N_o}$ is given in $\hat{U}$-$\hat{D}$ factor form. The objective is to obtain P_i^{No} in $\bar{U}$-$\bar{D}$ factor form, where $\hat{U}$, $\bar{U}$ are unit upper triangular matrices, and $\hat{D}$, $\bar{D}$ are diagonal matrices.
2. Apply the Modified Gram-Schmidt Time Update Algorithm to the following expression:

$$UDU' = \tilde{Q}_i + \tilde{A}_i'^{N_o}(K_i) P_{i+1}^{N_o} \tilde{A}_i'^{N_o}(K_i) \tag{3.17}$$

   where U is upper triangular and D is diagonal.
3. Define $v = \tilde{A}_i'^{N_o}(K_i) P_{i+1}^{N_o} \tilde{B}_i^{N_o}$ and $\alpha = R + \tilde{B}_i'^{N_o} P_{i+1}^{N_o} \tilde{B}_i^{N_o}$, we then have

$$P_i^{N_o} = \overline{UDU}' = UDU' - \alpha^{-1} vv' \tag{3.18}$$

4. Apply rank-one update to equation (3.18) to obtain the $\bar{U}$-$\bar{D}$ factor for $P_i^{N_o}$.

The factorization algorithms used in the previous procedure are, for completeness's sake, summarized in Appendix C. The computer coding to implement these algorithms are listed in Appendix D. It should be noted that the computer coding used during the simulation work has not made full use of the efficiency which is inherent in the factorization techniques, since the efficient coding of the algorithms may result in programs which are unreadable and relatively complicated to debug. Nonetheless, it should be remembered that factorization techniques do have the characteristic of computational efficiency and the advantage of smaller computer storage requirement.

CHAPTER 4

ESTIMATOR DESIGN

The design of the controller is based on the assumption that the estimates for the unknown parameters in the matrices A, B are provided by an estimator. It is the objective of this chapter to develop an estimator for the parameters of this magnetic ball suspension system.

4.1 Extended Kalman Filter

We investigate the use of the Kalman filter [7,11,15] to estimate the two unknown parameters θ_1 and θ_2 in the system. We augment the two parameters as two additional states to the system of equation (3.8). The resulting system is an eighth order system

$$y_{k+1} = \begin{bmatrix} \xi(k+1) \\ \theta_1(k+1) \\ \theta_2(k+1) \end{bmatrix} = \left[\begin{array}{c|cc} \tilde{A}_k^{N_o} & \multicolumn{2}{c}{0} \\ \hline 0 & 1 & 0 \\ & 0 & 1 \end{array}\right] \begin{bmatrix} \xi(k) \\ \theta_1(k) \\ \theta_2(k) \end{bmatrix} + \begin{bmatrix} \tilde{B}_k^{N_o} \\ 0 \\ 0 \end{bmatrix} u_k + \begin{bmatrix} \Gamma \\ 0 \\ 0 \end{bmatrix} w_k \tag{4.1}$$

together with observation

$$Z_k = \xi_k + \eta_k = Hy_k + \eta_k \tag{4.2}$$

where

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

and w_k, η_k are independent stochastic sequences with zero mean, and y_0, w_k, η_k are statistically independent Gaussian variables with variances

$$E\{y_o y_o'\} = \tilde{S}_o, \quad E\{\eta_i \eta_j'\} = M\delta_{ij}, \quad E\{w_i^2\} = V_w.$$

It should be observed that the observation equation (4.2) differs from the system's original observation equation (3.1) by a noise term η_k. This term is appended to the observation equation (4.2) in order to ensure that the Kalman filter to be designed has non-singular covariances. To reduce the deviation from the original model as much as possible, the covariances of this noise will be assumed to be extremely small. Specifically, the noise variance of η_k is taken to be 1/100000 times of the V_w, the noise variance of w_k.

With the two unknown parameters entering into the system as two additional state variables, the new system governed by equation (4.1) is a non-linear system, thus calling for the application of an extended Kalman filter. The operation of this extended Kalman filter for the magnetic ball suspension system can be described by the following sets of equations:

<u>Measurement Update</u>:

$$\hat{y}_{k+1} = \tilde{y}_k + C_k(z_k - H\tilde{y}_k) \tag{4.3}$$

$$\tilde{y}_0 = E\{y_0\} \text{ given}$$

$$C_k = \tilde{S}_k H'(H\tilde{S}_k H' + M)^{-1} \tag{4.4}$$

$$S_k = \tilde{S}_k - C_k H\tilde{S}_k \tag{4.5}$$

where $\tilde{S}_o$ is given as $\tilde{S}_o = \text{diag}\{vn,vn,vn,vn,0.1,0.1\}$, and $vn = \tilde{y}_o/10$.

Time Update:

$$\tilde{y}_{k+1} = \phi_k \hat{y}_k \tag{4.6}$$

$$\tilde{S}_{k+1} = \Psi_k S_k \Psi_k' + V_w G_k G_k' \tag{4.7}$$

where ϕ_k is the transition matrix of equation (4.1) with the feedback control incorporated, that is,

$$\phi_k = \left[\begin{array}{c|cc} \tilde{A}_k^{N_o} & \multicolumn{2}{c}{0} \\ \hline 0 & 1 & 0 \\ & 0 & 1 \end{array}\right] + \begin{bmatrix} \tilde{B}_k^{N_o} K(k) \\ 0 \\ 0 \end{bmatrix}$$

Ψ_k is the linearized system transition matrix, that is,

$$\Psi_k = \left.\frac{\partial \phi_k}{\partial y_k}\right|_{y_k = y_k^N} \tag{4.8}$$

where $y_k^N = (y_k^N(1), y_k^N(1), y_k^N(2), \ldots, y_k^N(8))'$ is the nominal trajectory for the filter and is obtained from

$$y_{k+1}^N = \phi_k y_k^N \tag{4.9}$$

$$y_0^N = \begin{bmatrix} \xi(N_o) \\ \theta_1(N_o) \\ \theta_2(N_o) \end{bmatrix}$$

and G_k is the augmented matrix for the noise term w_k

$$G_k = \begin{bmatrix} \Gamma \\ 0 \\ 0 \end{bmatrix} \tag{4.10}$$

From equation (4.8), it is found that all the entries of the matrix Ψ_k are the same as the corresponding entries of the matrix ϕ_k except the following ones:

$$\Psi_j(2,7) = y_j^N(1)$$

$$\Psi_j(2,8) = -\sum_{i=1}^{6} k_i(j)y_j^N(i)$$

$$\Psi_j(4,7) = y_j^N(3) + y_j^N(4)$$

$$\Psi_j(4,8) = -k_3(j)y_j^N(3) - k_4 y_j^N(4)$$

$$\Psi_j(6,7) = y_j^N(5) + y_j^N(6)$$

$$\Psi_j(6,8) = -k_5(j)y_j^N(5) - k_6(j)y_j^N(6)$$

where the k_i's are the control feedback gains defined in equation (3.16).

With the estimator and controller having been developed, the design of the entire magnetic ball suspension control system is complete. A block diagram for the discrete linearized plant with the SAFER controller and the extended Kalman filter is shown in Fig. 3.

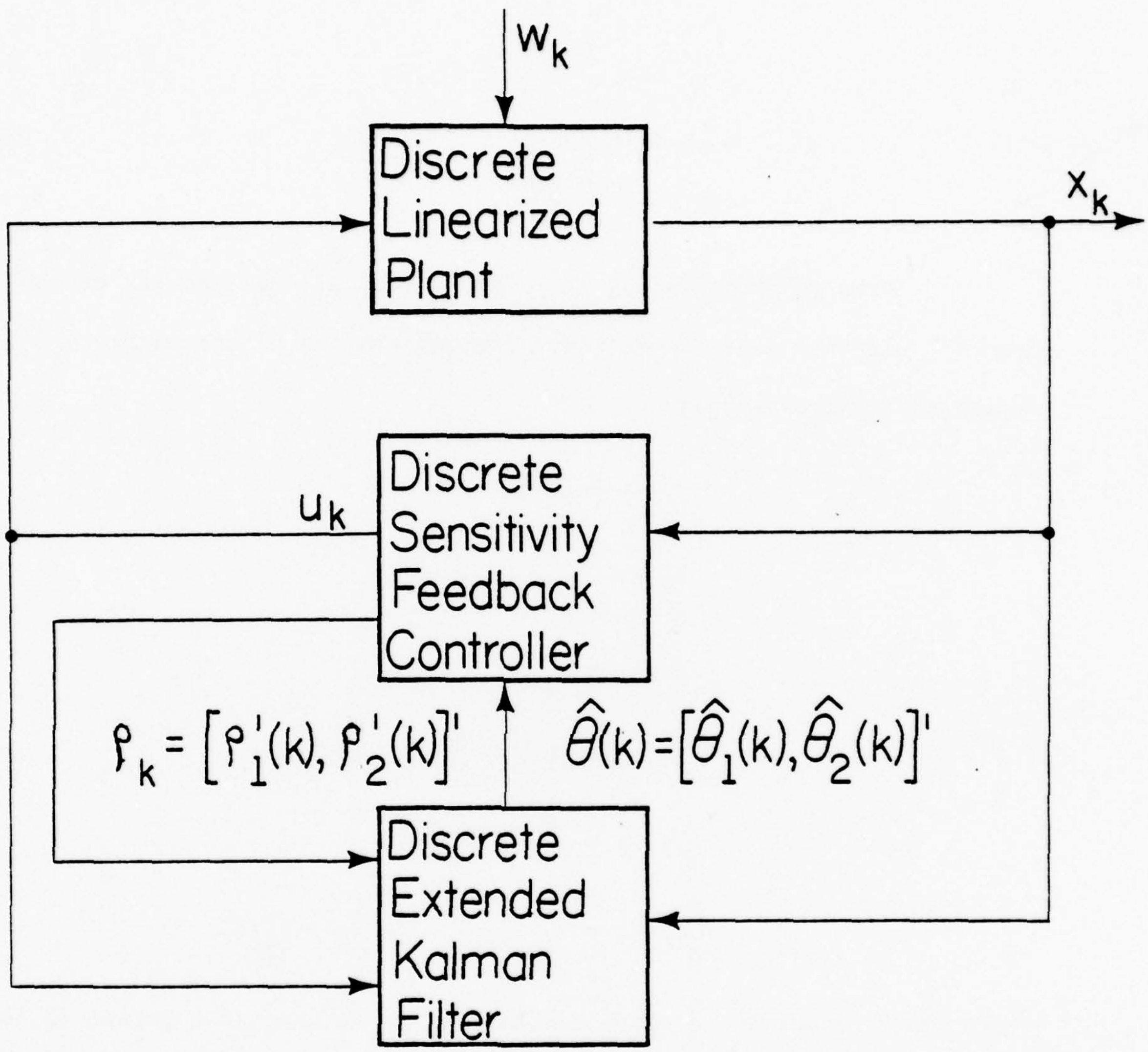


Fig. 3. Closed-Loop Magnetic Ball Suspension System

4.2. Filter Factorization

As mentioned in Section 3.2 of this thesis, the factorization method yields numerically stable and efficient computations for matrix algorithms. The conventional Kalman filter equations as stated in Section

4.1 are particularly prone to numerical instability [3,4,8]. In fact, it is the experience of the author that during the simulation work done on the DEC-10 computer, computation underflow occurred during the matrix inversion for equation (4.4). To avoid this matrix inversion, equations (4.3) through (4.5) are cycled through once for each of the six components of the observation vector z_k. In this manner, the matrix inversion in equation (4.4) reduces to a simple scalar division. Nevertheless, the underflow problem remained, probably due to other matrix manipulations in equations (4.3-4.5). It was only after the entire Kalman filter was constructed using factorization techniques that this problem of computation underflow vanished.

The Kalman filter factorization algorithm [2,3] is given in Appendix C and the corresponding computer coding is given in Appendix D. It should be noted that in the time update equation of the estimate for this magnetic ball suspension system, a non-linear transition matrix ϕ_k is used and the matrix Ψ_k used in the covariance propagation is the partial of ϕ_k evaluated along the nominal trajectory described by the equation (4.9).

CHAPTER 5

COMPUTATION RESULTS

5.1. Simulation of the Closed Loop System

Monte Carlo type of simulation is carried out on the digital computer, with the linearized discrete system governed by equation (3.8) considered as the plant. Investigation into the characteristics of the magnetic ball suspension system is done off line through the representation of the system by this plant. The sampling period for the system is chosen to be 0.01 second, which corresponds to 100 Hertz. However, the choice of a sampling rate is rather artificial since the real time linearized (continuous) plant is not implemented. The reason for studying the discrete linearized plant off line only is that the computation for each of the discrete control signals takes approximately seven seconds for the DEC-10 computer, which makes the control of the magnetic ball suspension system in real time unfeasible. The real time magnetic ball system will be driven to instability long before the first control signal is applied. A real time control algorithm based on this sensitivity approach for this particular system will require a faster computation device, more efficient implementation of the algorithm through programming techniques, and probably a simplified model of lower dimension also. Hence, the simulation work done can be thought of primarily as an exercise to study a certain stochastic system under the application of the sensitivity approach control algorithm, bearing in mind that this system bears close resemblance to the magnetic ball suspension system and it may actually represent the magnetic ball system if some of the hindrance in implementations are overcome.

Going back to the mechanics of the simulation, only forty iterations are carried out for each set of conditions investigated as it is relatively time consuming for the computation of the control at each discrete instant. With a sampling period of 0.01 second, the forty iterations correspond to a time span of 0.4 second. This time interval should be quite sufficient for avoiding loss of stability of the system, based on experience obtained from real time control experiments done on the analog version of the magnetic ball system at the University of Illinois and based also on some hindsight after the digital simulation. A flow chart illustrating the various steps involved in the simulation process is given in Fig. 4. The plant under consideration has the sensitivity states incorporated into it and is represented by equation (3.8). If the control is to be done in real time, these sensitivity states will have to be generated by a separate mathematical model by some processor while the position and velocity, which constitutes the original state variables of the magnetic ball system, will be measured by some transducers.

5.2. Stability of System

It is well known that for a linear discrete time invariant system, asymptotic stability of the system is guaranteed if there exists a positive definite solution to the time invariant matrix Riccati equation (3.12). See [1] for example. Since solving for such a solution may be extremely complicated and involved, an alternative which may provide a general indication of the system's stability is sought. In this magnetic ball suspension system,

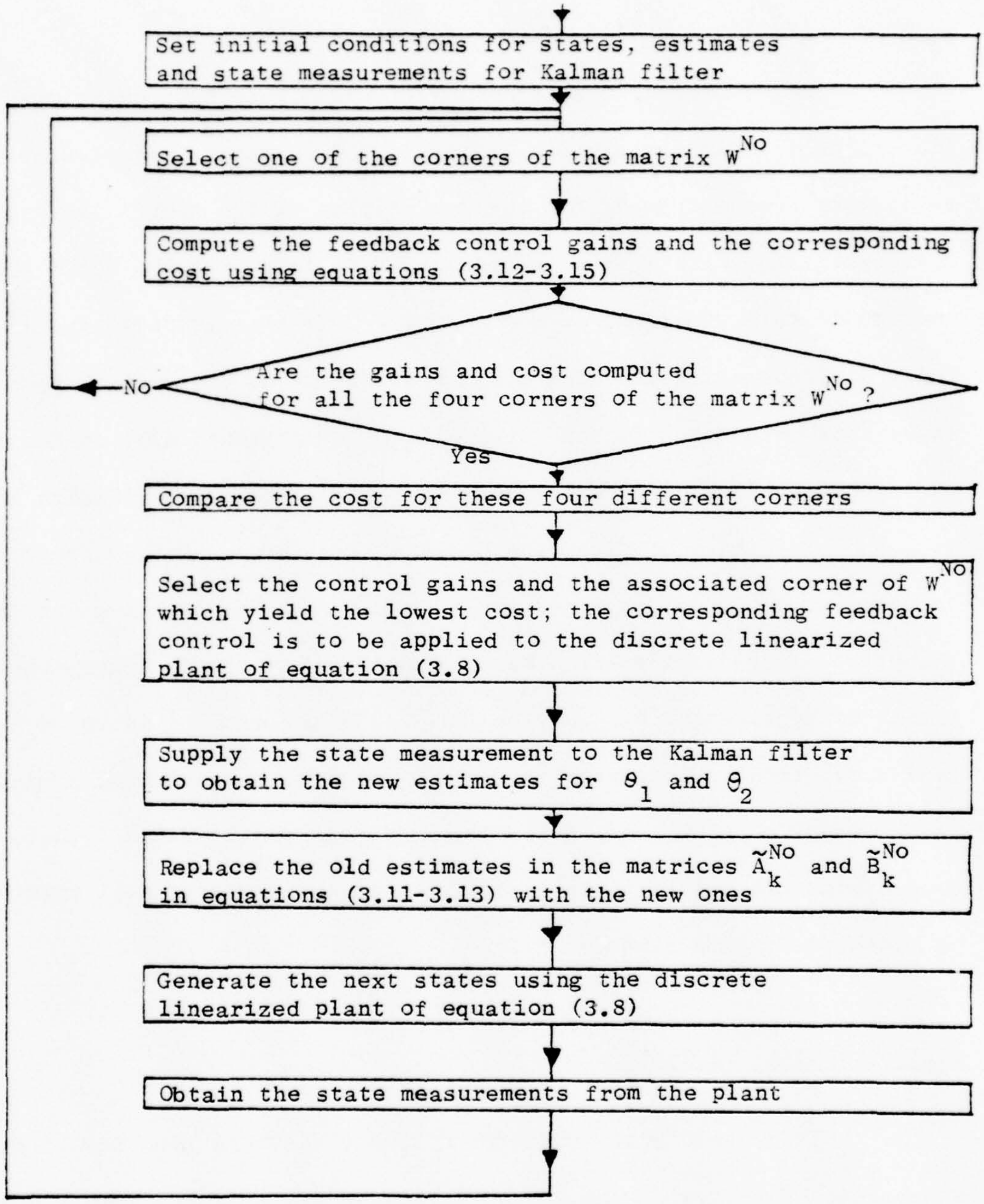


Figure 4 Flow chart for simulation

a rough idea concerning the stability of the system is determined by checking the eigenvalues of the system's transition matrix with feedback incorporated. That is, the eigenvalues of the matrix $(\tilde{A}_k^{N_o}(K_k) - \tilde{B}_k^{N_o}K_k)$ in equation (3.15) are checked. If the eigenvalues are within the unit circle, asymptotic stability is supposed to be satisfied. Indeed, from the experience obtained from the simulation work, if the eigenvalues are outside the unit circle, the system will diverge. In cases where the eigenvalues are within the unit circle, the states do decay to zero. Hence, this method does provide a good approximate indication of the stability of the system.

5.2.1. Effect of Weighting Matrices on Stability

The choice of the various weighting matrices Q, R, λW^{N_o} and the number of stages to be optimized ν, will influence the location of the eigenvalues of the system governed by equation (3.8). Certain choices of these weighting matrices will result in a system with three eigenvalues located far apart from the other three. That is, there is a set of three slow eigenvalues and also another set of fast eigenvalues for the system. This presence of slow and fast time constants in the system results in an unsatisfactory control performance. The position of the ball, for instance, will return to the nominal position very slowly since the velocity of the ball approaches zero much faster than the position's deviation approaching zero. This kind of situation is generally undesirable for such a system. It is found that a choice of the matrix Q with a large weight on the position and a comparatively small weight on the velocity will yield a system with all six eigenvalues close by. Moreover, a small value for R will result in fast eigenvalues corresponding to the velocity and its sensitivity states. A large value for λ will also result in the same situation. It should be noted that a small value for λ corresponds to a bigger estimation budget, which should enhance the accuracy for the estimates. However, a λ value which is too small may drive the system into instability. Along side with small λ, a more evenly distributed element values of the matrix W^{N_o} will result in faster eigenvalues for the velocity and its corresponding sensitivity states. (The matrix $W^{N_o} = \text{diag}(0.4, 0.2, 0.2, 0.2)$ is considered to have its weights more evenly distributed than $W^{N_o} = \text{diag}(0.7, 0.1, 0.1, 0.1)$). Finally, increasing the number of stages of optimization ν, will enhance the stability of the system.

Only after numerous experiments with the system, considering both the stability of the system as well as the accurate estimation of parameters, were the values of the weighting matrices chosen. The values for some of them are listed in Table 2. The various choices of the matrix W^{N_o} are listed in Appendix B.

Table 2

Values for some of the weighting matrices

$$Q = \begin{bmatrix} 400 & 0 \\ 0 & 0.2 \end{bmatrix} \qquad \lambda = 1 \qquad R = 0.5$$

$$\nu = 5$$

5.3. Simulation Results

The behavior of the system was explored by running the system under various conditions. Noise was injected into the system according to the system equation (3.8) using a programmed random noise generator. The various sets of condition applied to the system are listed in Table 3. All the initial sensitivity states are assumed to be zero. For condition (15), the actual parameters in the plant are changed at $t = 0.2$ seconds. For condition (16), in addition to the changing of parameters at $t = 0.2$ seconds, the position and velocity states are also perturbed. These cases are used to investigate the response of the system with time varying parameters. This will be further discussed later in the thesis.

The time response of the states x_1 (position deviation) and x_2 (velocity deviation) are shown in Figures (5.1a-5.16a). The error of the estimates $e_1 = \theta_1 - \hat{\theta}_1$ and $e_2 = \theta_2 - \hat{\theta}_2$, where $\hat{\theta}_1$ and $\hat{\theta}_2$ are the estimates for θ_1 and θ_2 respectively, are shown in Figures (5.1b-5.16b). The control response corresponding to conditions (5), (6), (7), (15), and (16) are also shown in Figures 5.5c, 5.6c, 5.7c, 5.15c and 5.16c respectively.

5.3.1. Region of Stability

In conditions (1) through (4), the initial position deviation is set at +/-0.05 m., which is a 25% deviation from the nominal value of 0.2 m.. As shown in Figures (5.1a-5.4a), the states invariably decay to zero. Hence, a reasonably wide region of stability can be inferred from these simulation results.

Table 3 Initial conditions and parameter's values for simulation

condition			Parameters				Noise variance of w
			Plant values		Initial estimates		
	$x_1(0)$	$x_2(0)$	θ_1	θ_2	$\hat{\theta}_1(0)$	$\hat{\theta}_2(0)$	V_w
1	0.05	0.05	0.5	-0.4	0.485	-0.42	0.01
2	-0.05	0.05	0.5	-0.4	0.485	-0.42	0.01
3	0.05	-0.05	0.5	-0.4	0.485	-0.42	0.01
4	-0.05	-0.05	0.5	-0.4	0.485	-0.42	0.01
5	0.005	0.05	0.5	-0.4	0.485	-0.42	0.01
6	0.005	0.05	0.5	-0.4	0.485	-0.42	0.5
7	0.005	0.05	0.5	-0.4	0.485	-0.42	0.00055
8	0.005	0.05	0.5	-0.3	0.485	-0.42	0.01
9	0.005	0.05	0.5	-0.5	0.485	-0.42	0.01
10	0.005	0.05	0.4	-0.4	0.485	-0.42	0.01
11	0.005	0.05	0.6	-0.4	0.485	-0.42	0.01
12	0.005	0.05	0.6	-0.3	0.485	-0.42	0.01
13	0.005	0.05	0.4	-0.5	0.485	-0.42	0.01
14	0.005	0.05	0.5	-0.4	0.5	-0.42	0.01
15	0.005	0.05	0.5	-0.4	0.485	-0.42	0.01
	at t = 0.02		0.4	-0.3			
16	0.005	0.05	0.5	-0.4	0.485	-0.42	0.01
	at t = 0.02 +0.005	+0.01	0.4	-0.3			

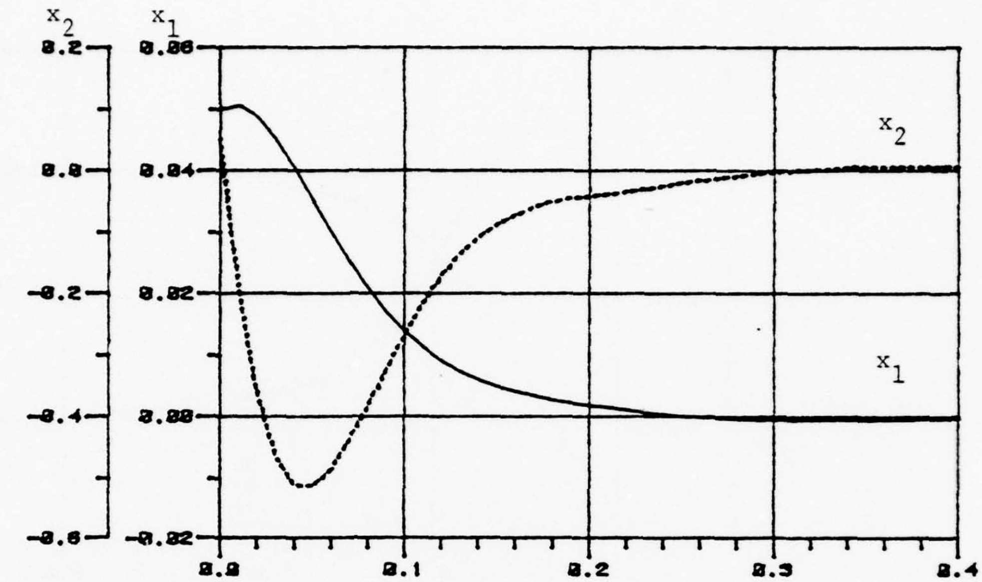


Figure 5.1a State response corresponding to condition (1)

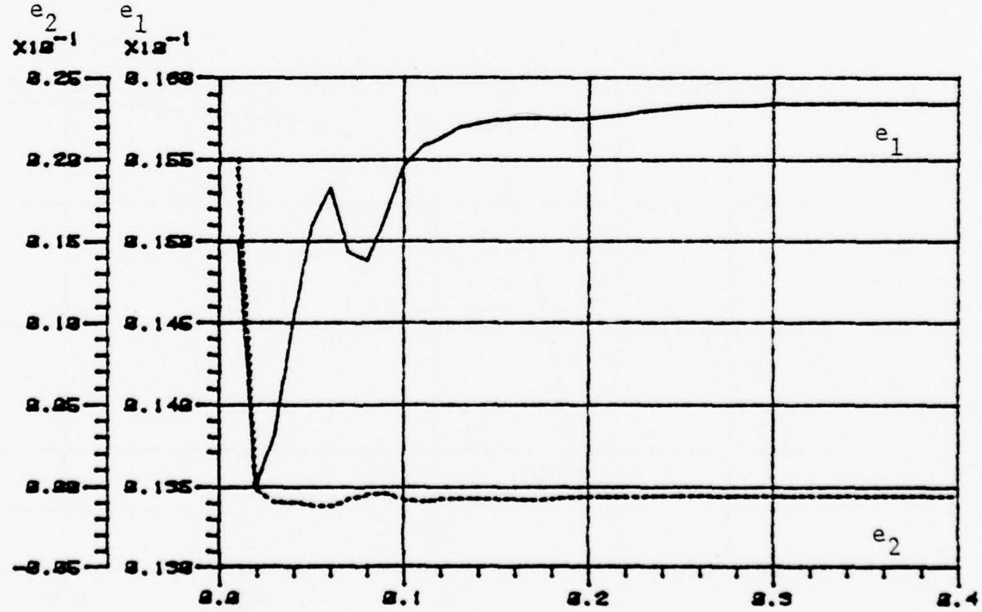


Figure 5.1b Time response of the estimates' error corresponding to condition (1)

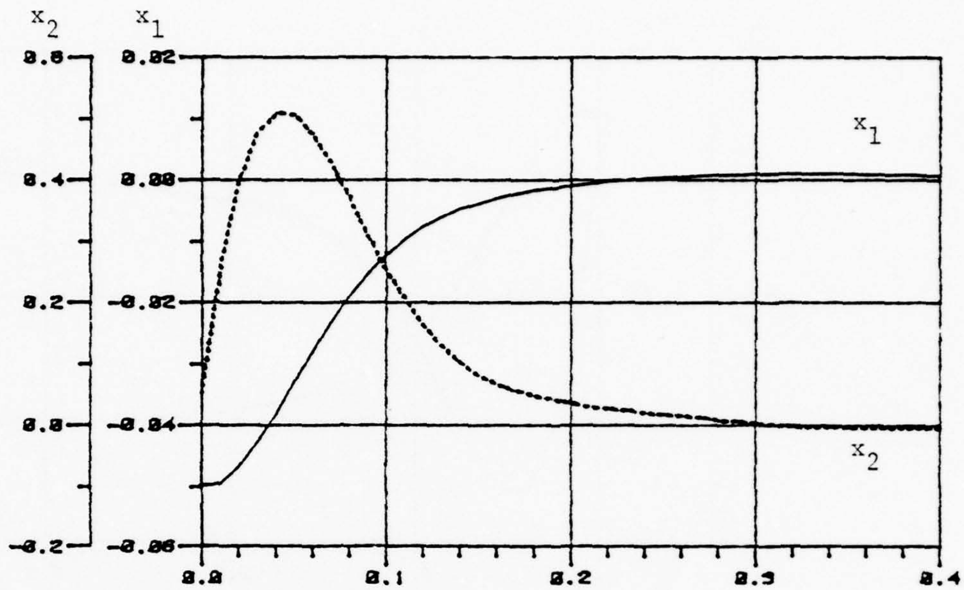


Figure 5.2a State response corresponding to condition (2)

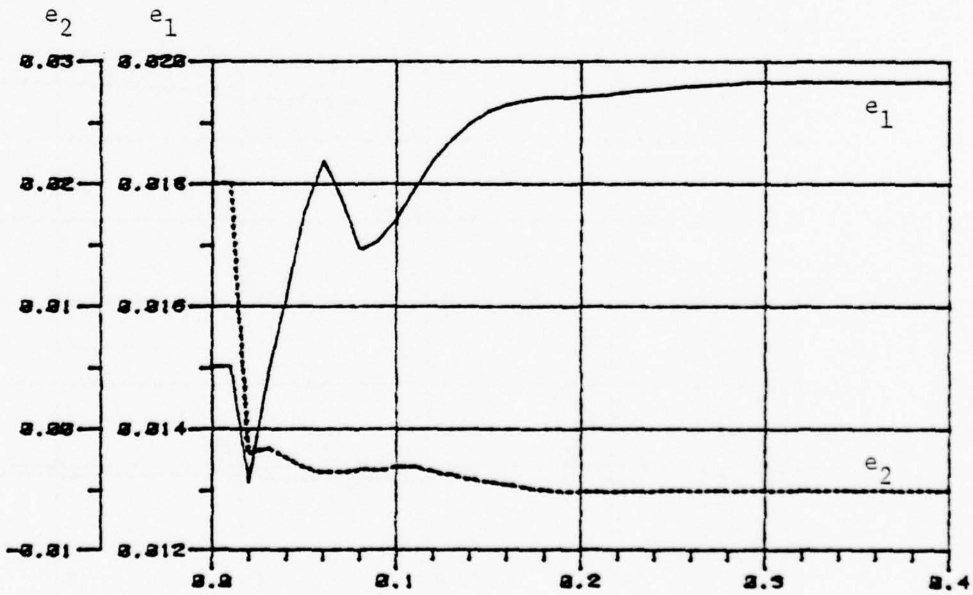


Figure 5.2b Time response of the estimates' error corresponding to condition (2)

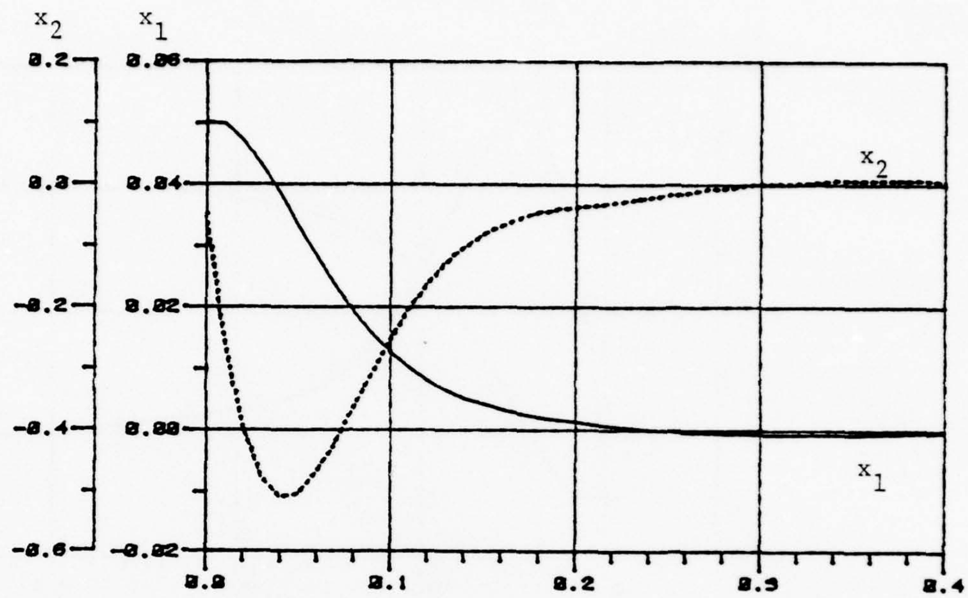


Figure 5.3a State response corresponding to condition (3)

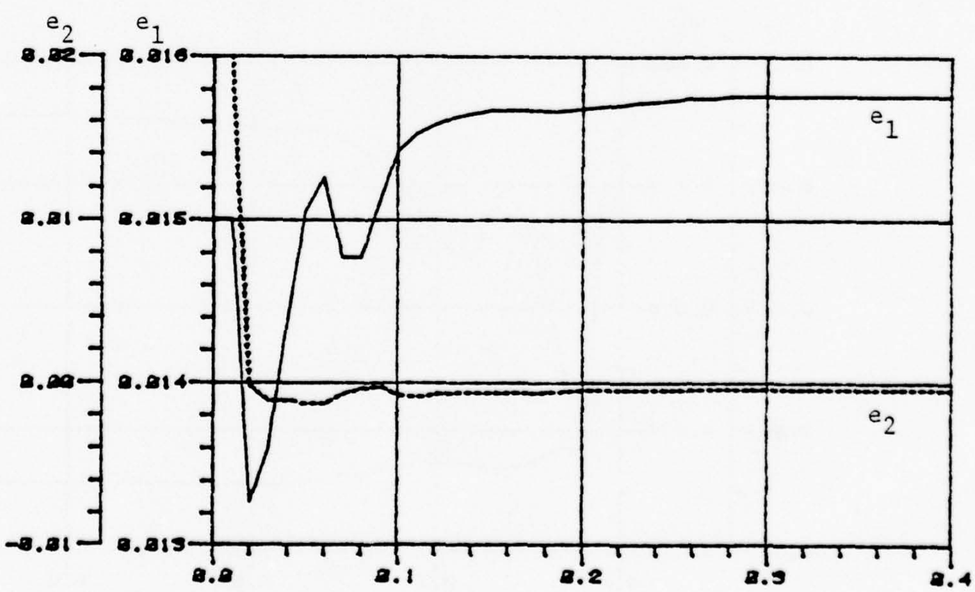


Figure 5.3b Time response of the estimates' error corresponding to condition (3)

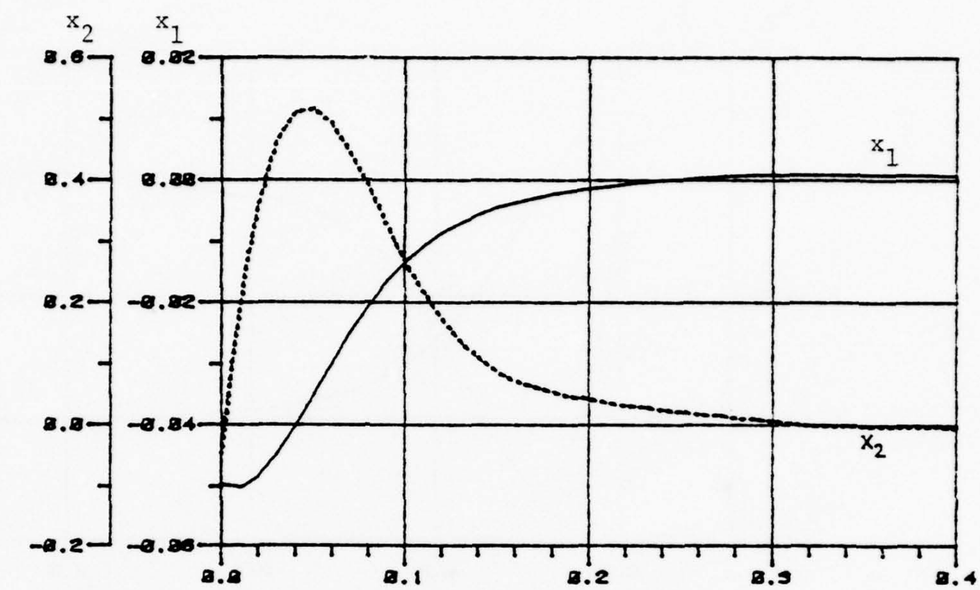


Figure 5.4a State response corresponding to condition (4)

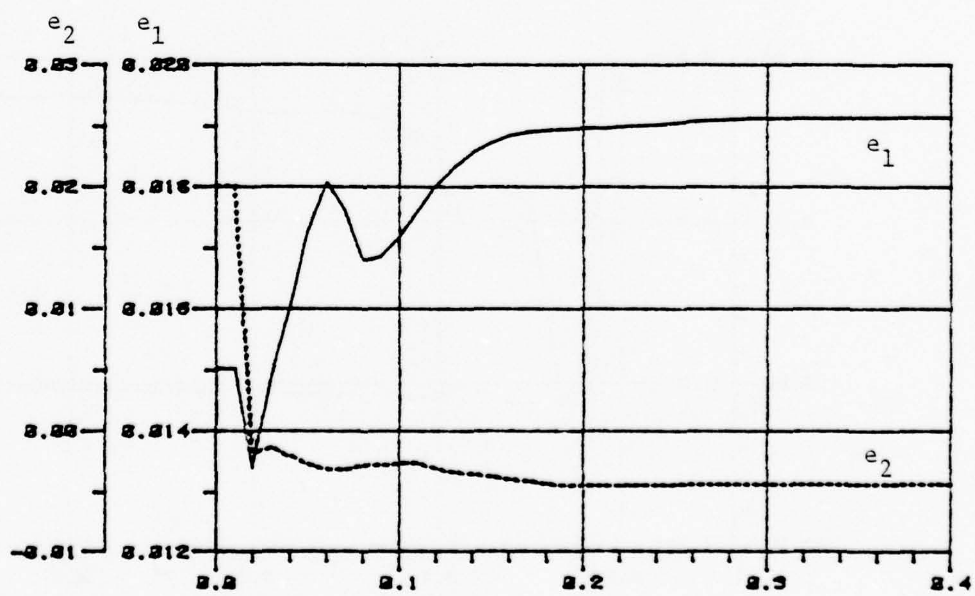


Figure 5.4b Time response of the estimates' error corresponding to condition (4)

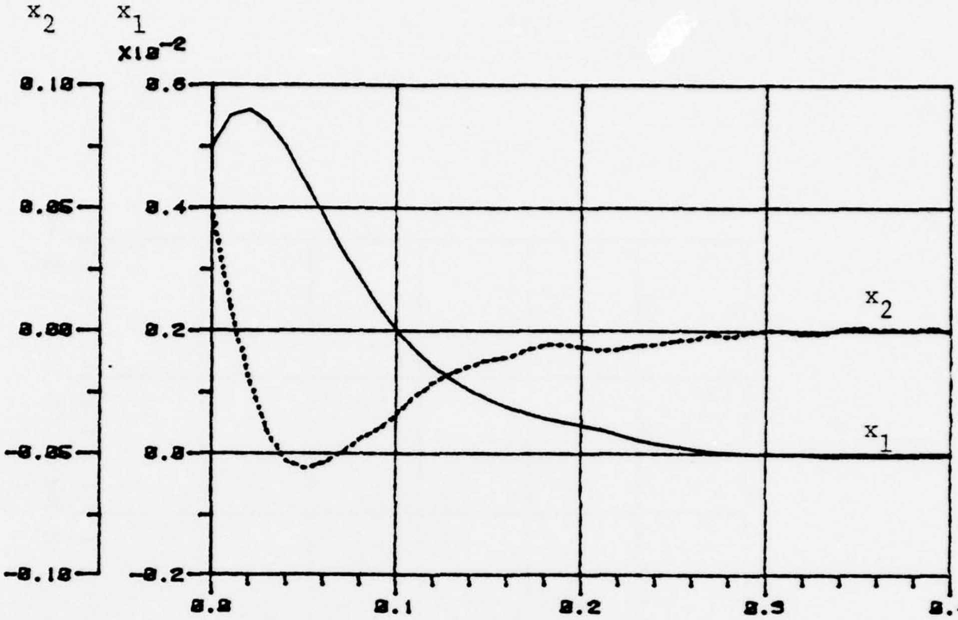


Figure 5.5a State response corresponding to condition (5)

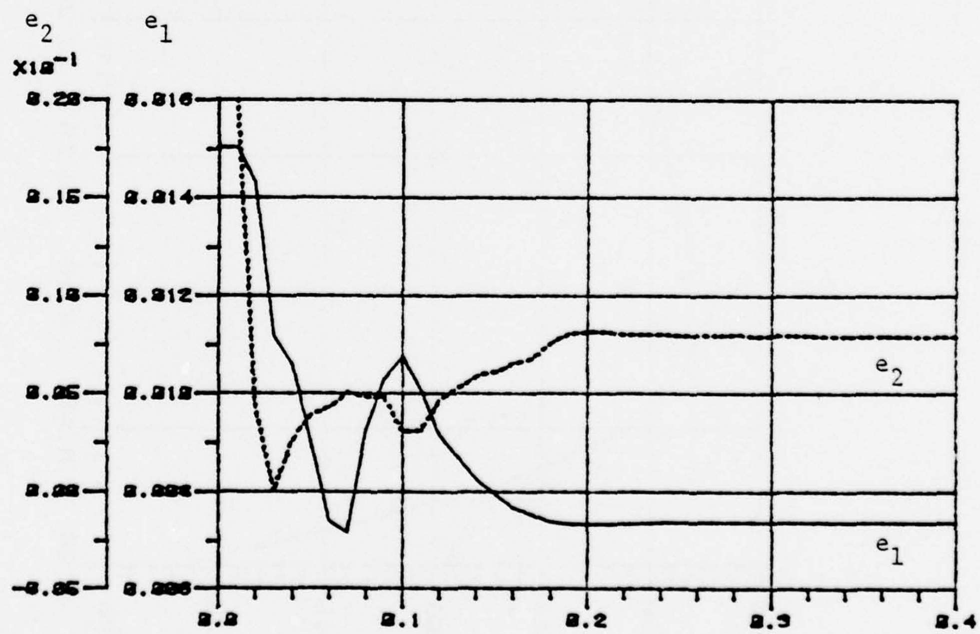


Figure 5.5b Time response of the estimates' error corresponding to condition (5)

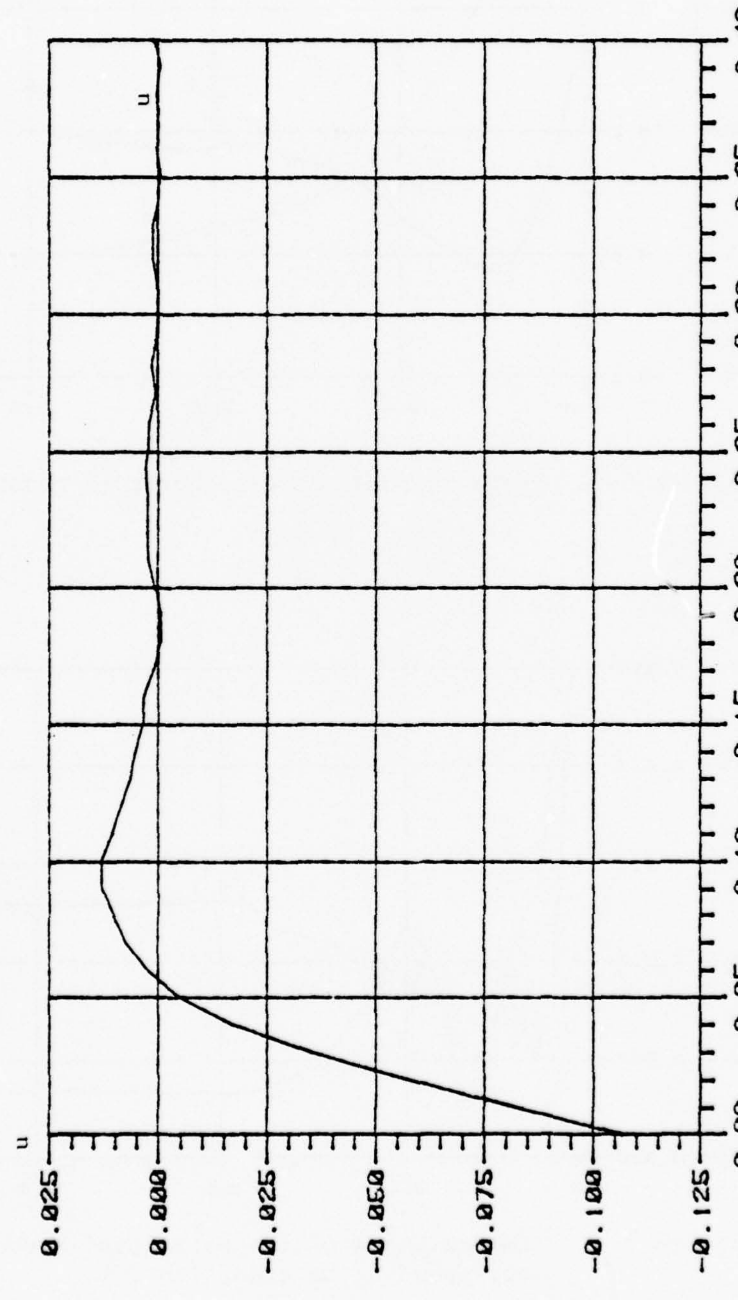


Figure 5.5c Control response corresponding to condition (5)

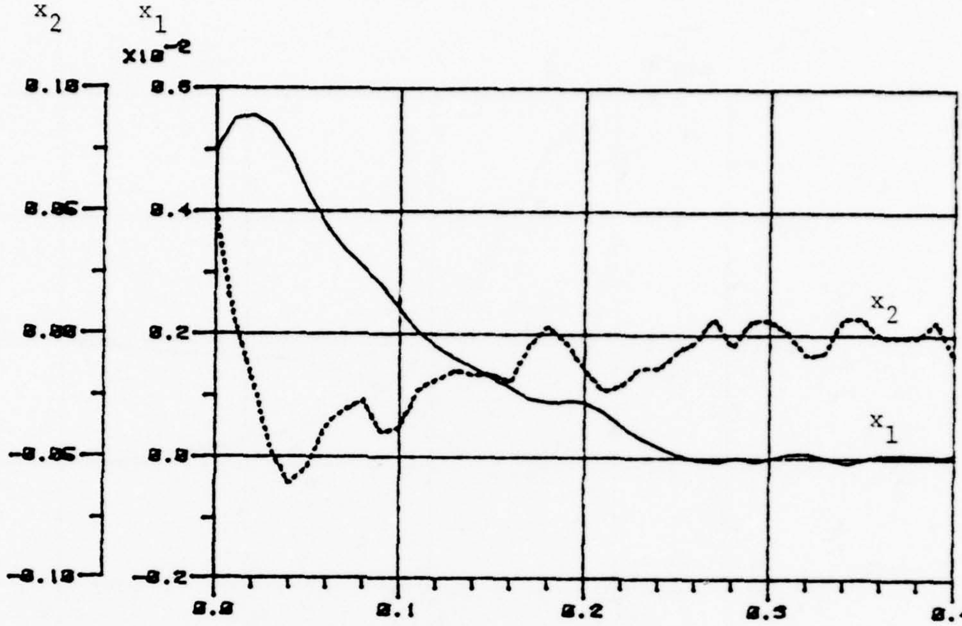


Figure 5.6a State response corresponding to condition (6)

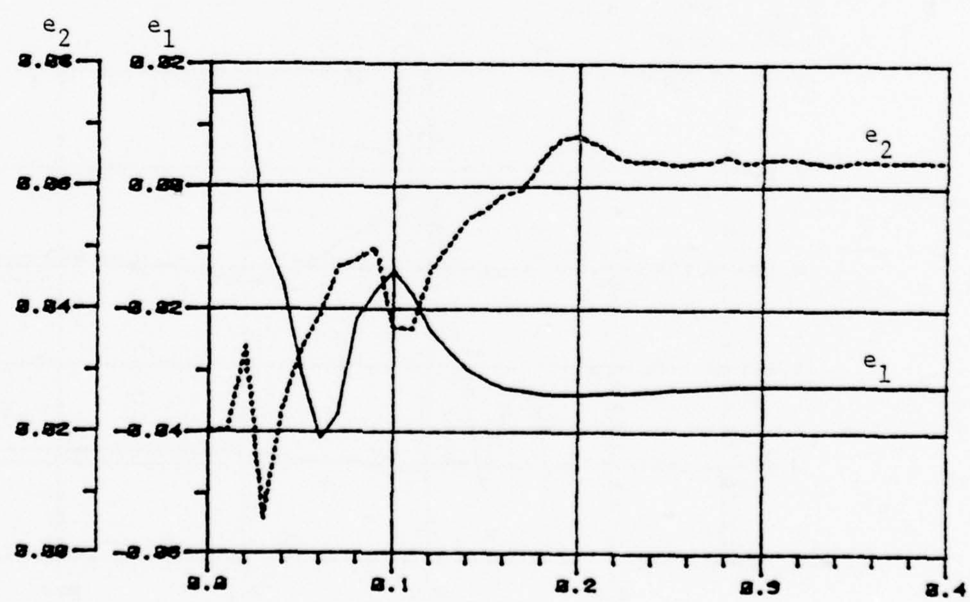


Figure 5.6b Time response of the estimates' error corresponding to condition (6)

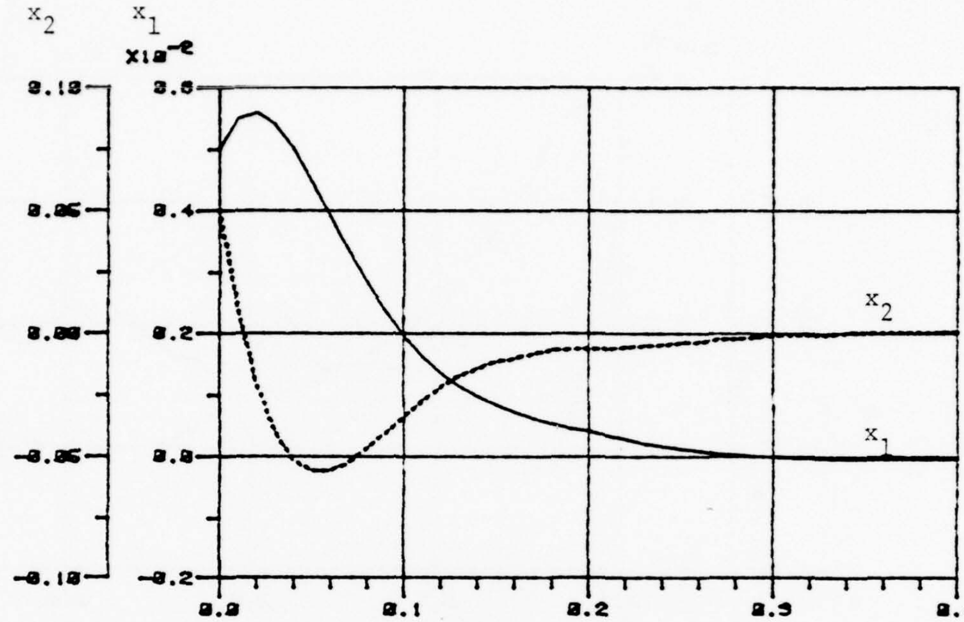


Figure 5.7a State response corresponding to condition (7)

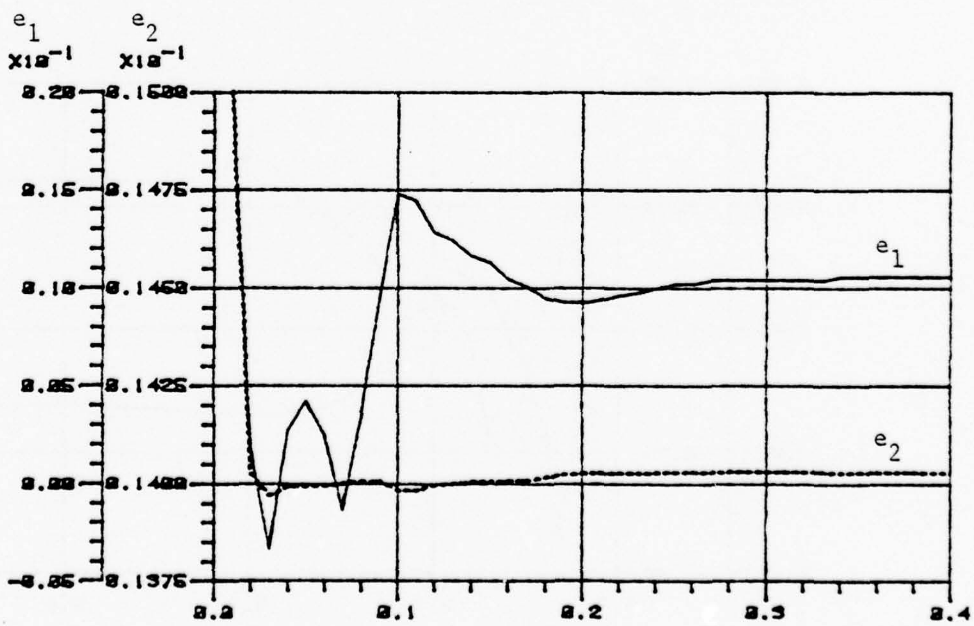


Figure 5.7b Time response of the estimates' error corresponding to condition (7)

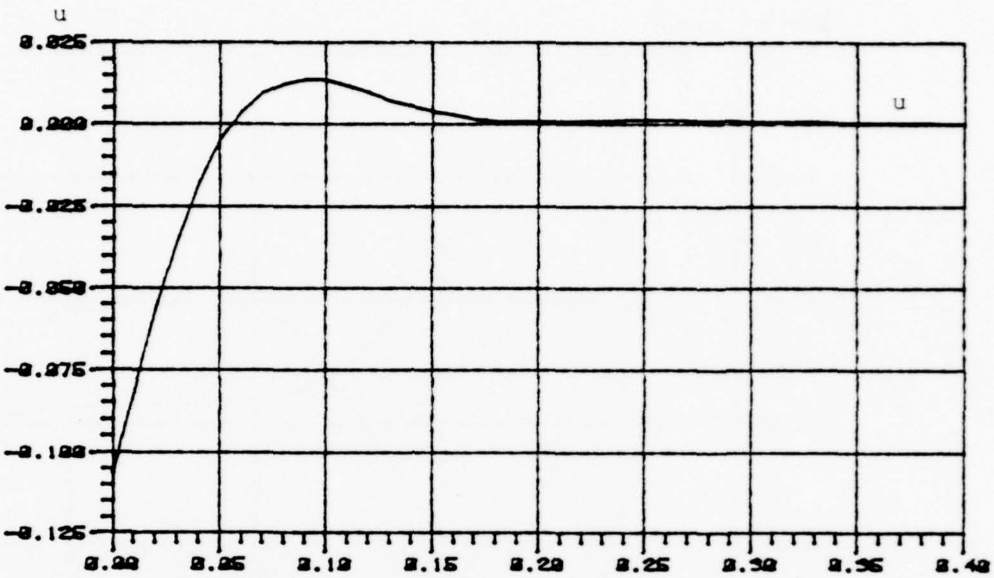


Figure 5.7c Control response corresponding to condition (7)

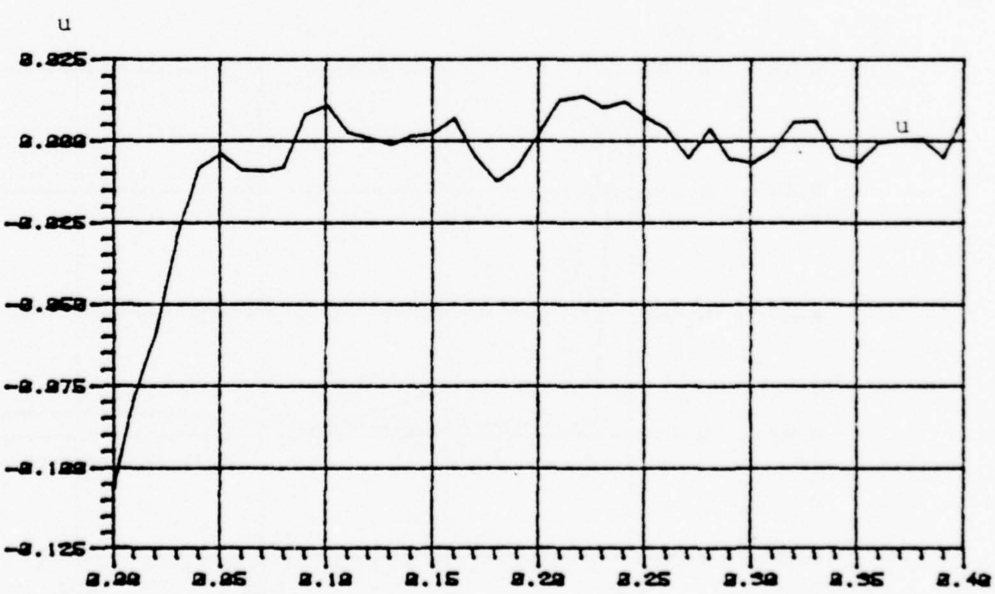


Figure 5.6c Control response corresponding to condition (6)

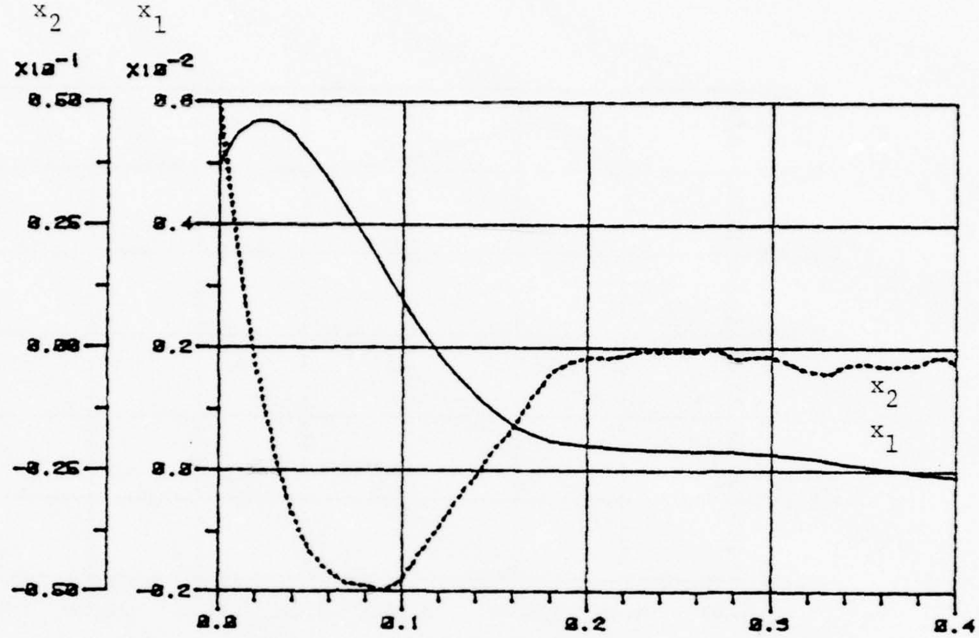


Figure 5.8a State response corresponding to condition (8)

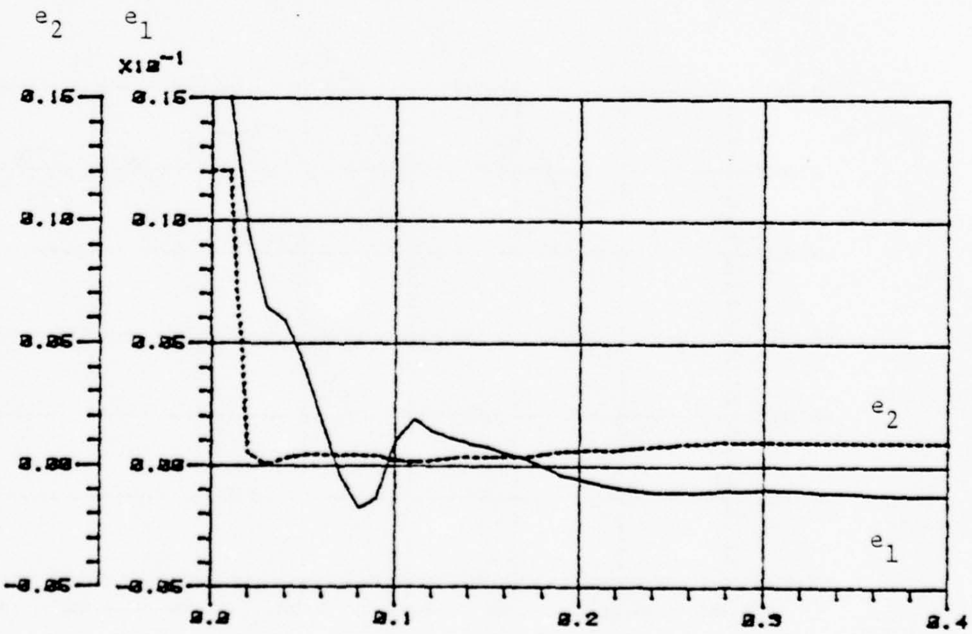


Figure 5.8b Time response of the estimates' error corresponding to condition (8)

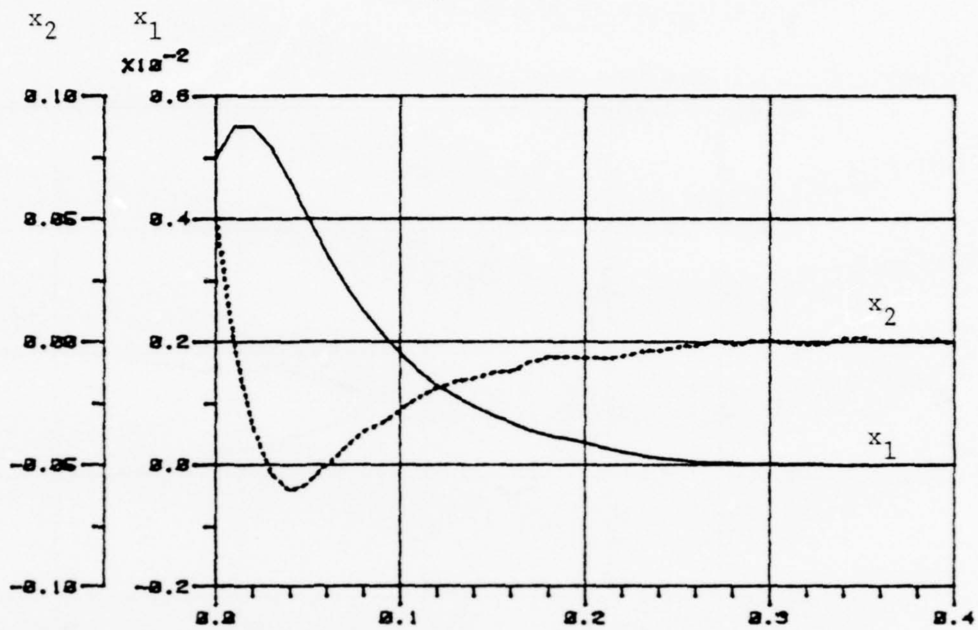


Figure 5.9a State response corresponding to condition (9)

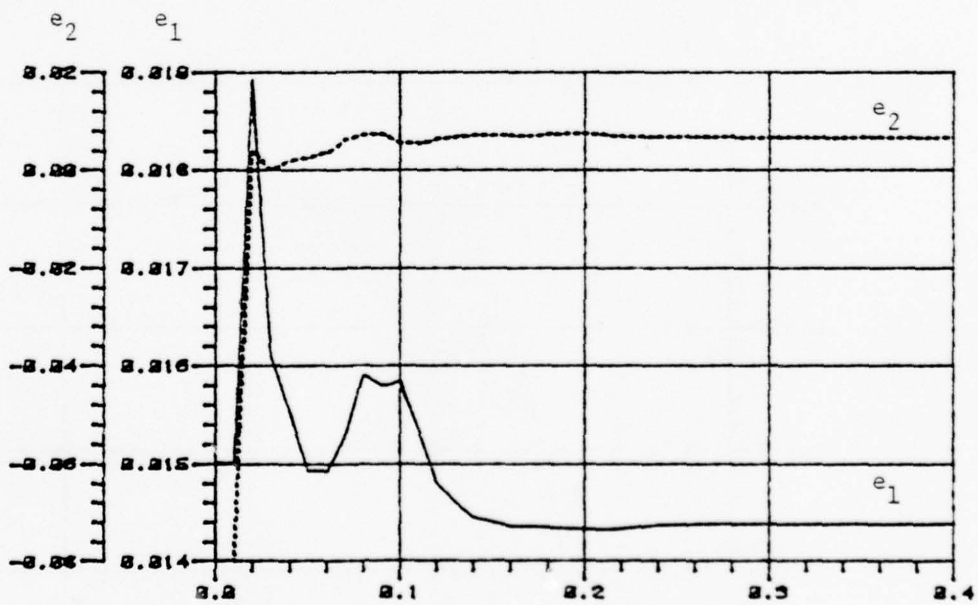


Figure 5.9b Time response of the estimates' error corresponding to condition (9)

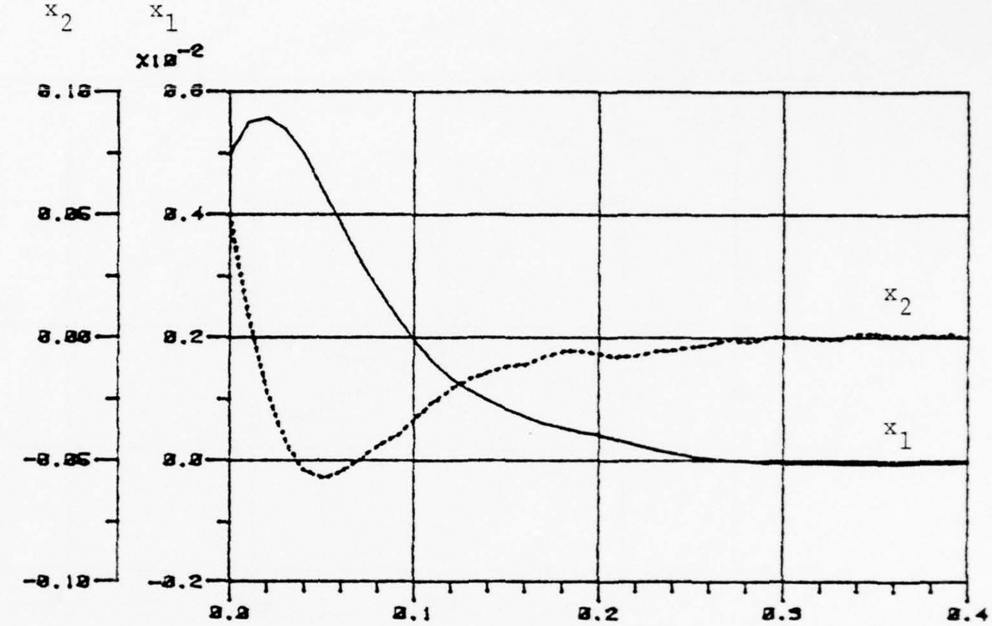


Figure 5.10a State response corresponding to condition (10)

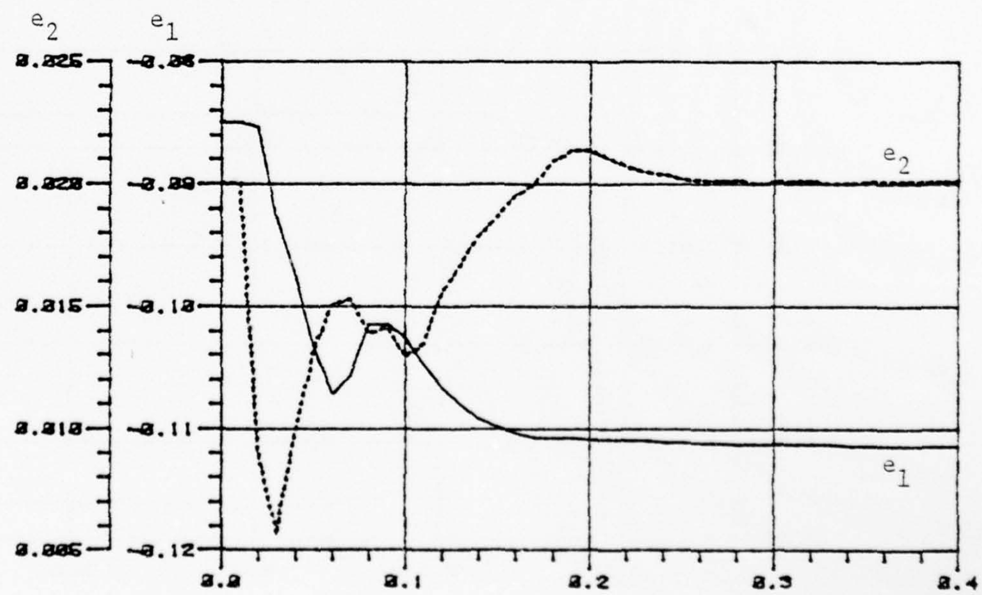


Figure 5.10b Time response of the estimates' error corresponding to condition (10)

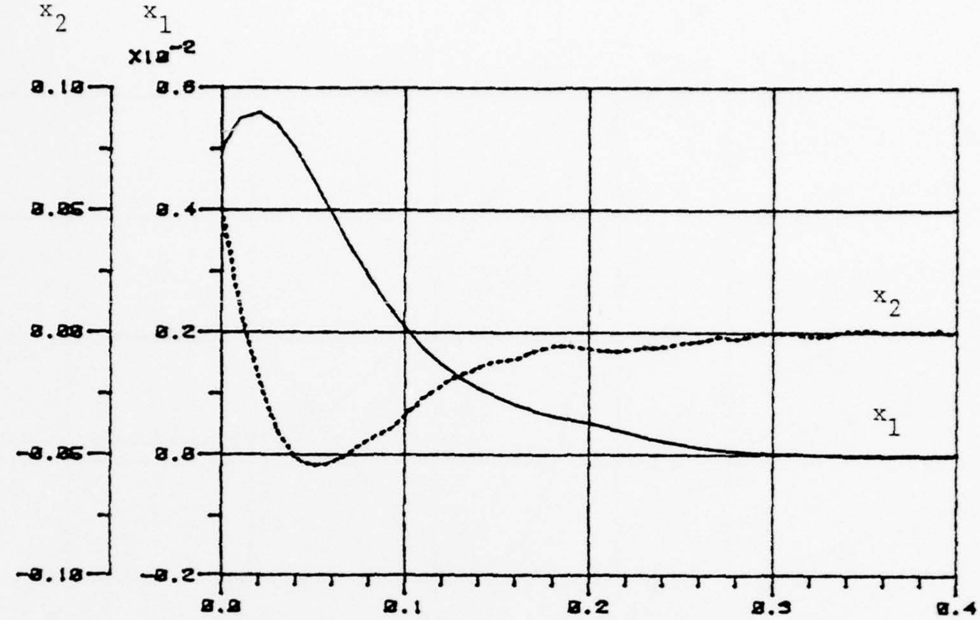


Figure 5.11a State response corresponding to condition (11)

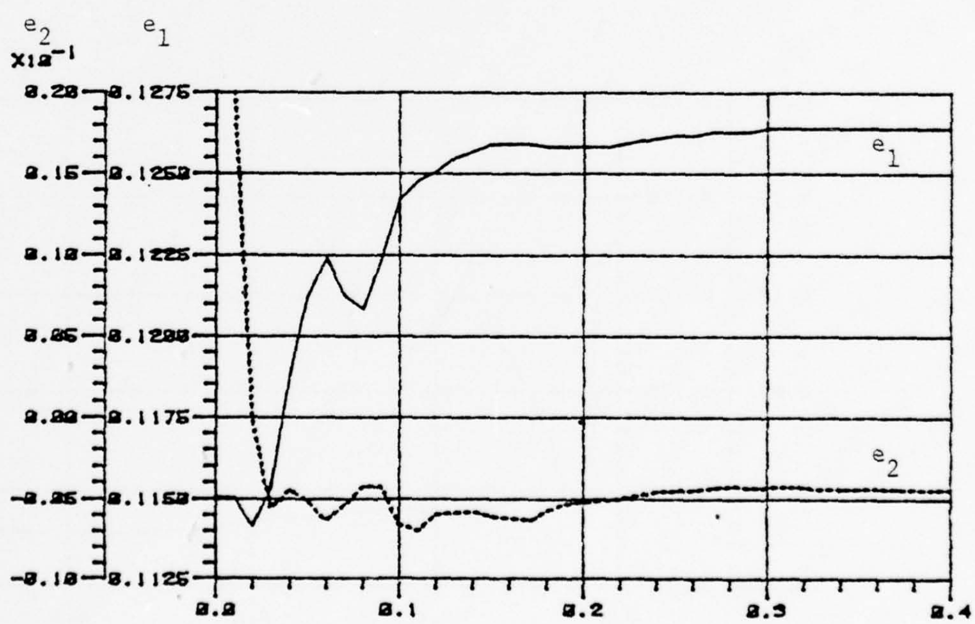


Figure 5.11b Time response of the estimates' error corresponding to condition (11)

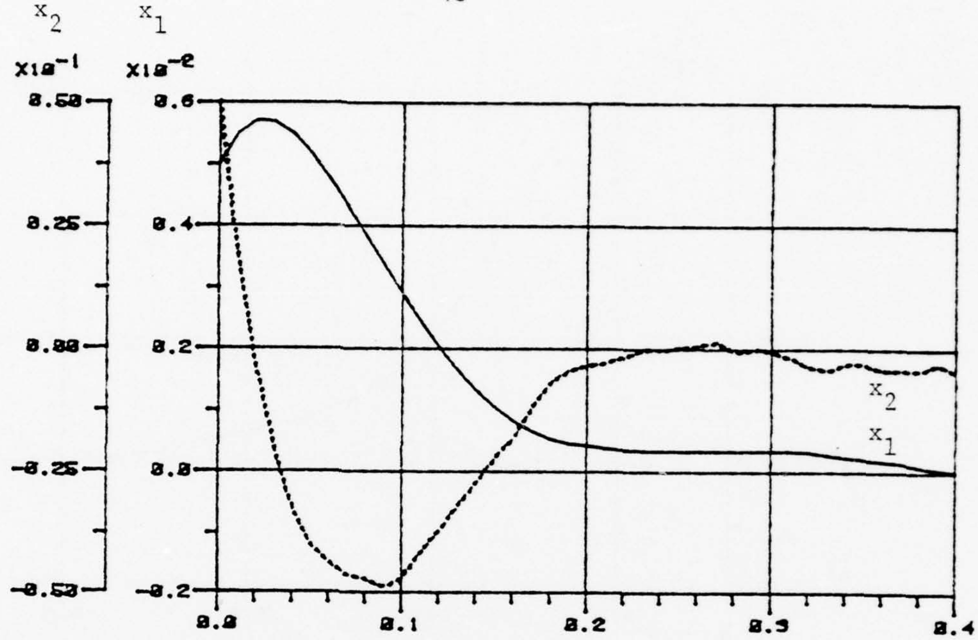


Figure 5.12a State response corresponding to condition (12)

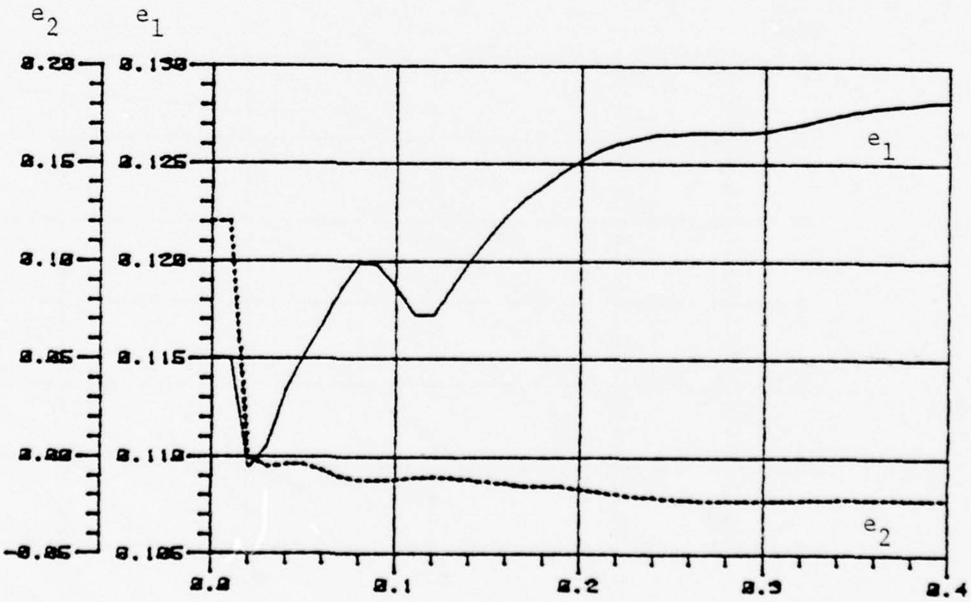


Figure 5.12b Time response of the estimates' error corresponding to condition (12)

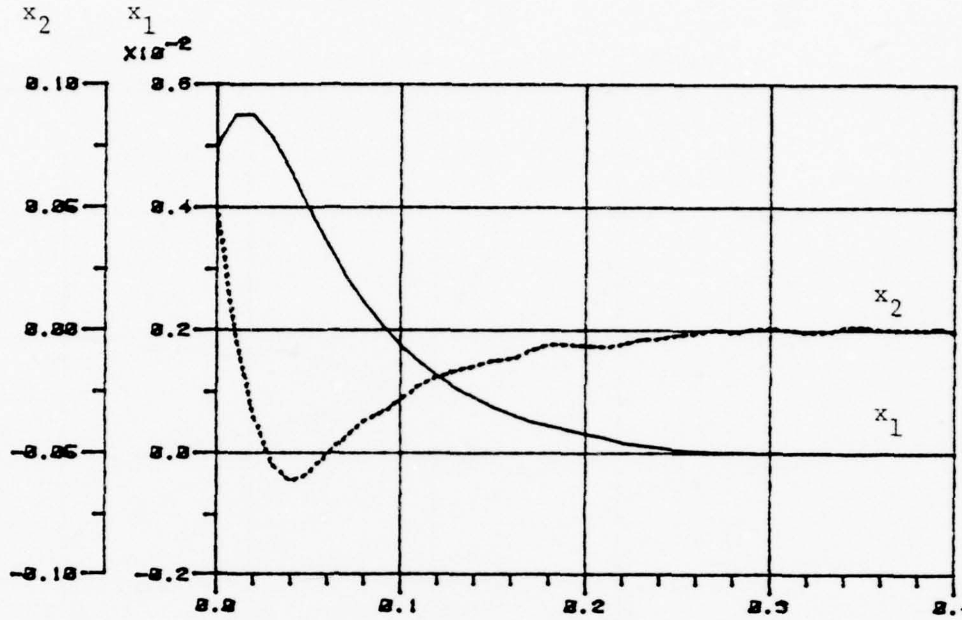


Figure 5.13a State response corresponding to condition (13)

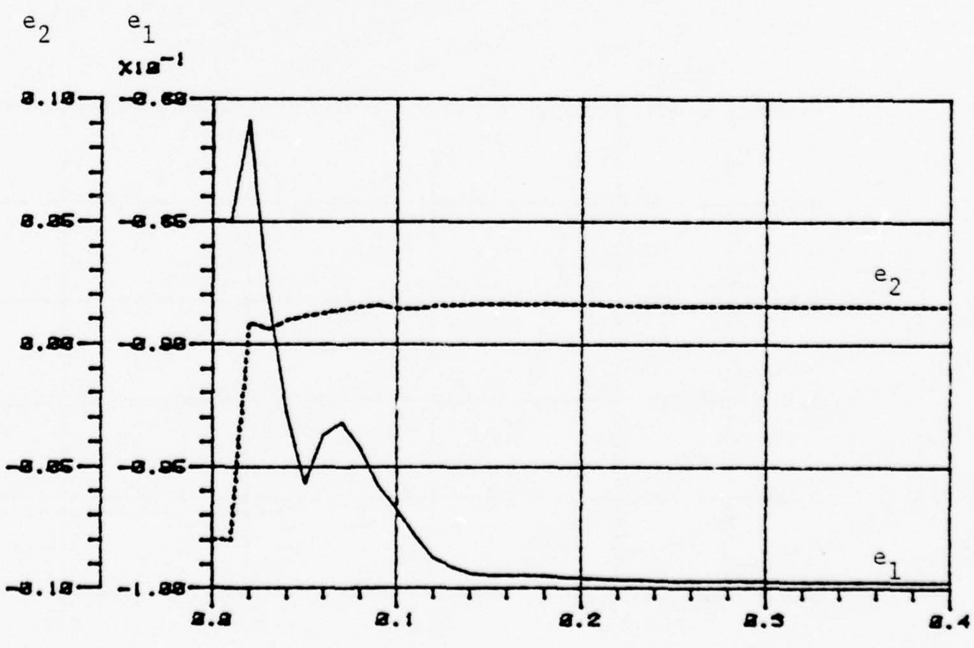


Figure 5.13b Time response of the estimates' error corresponding to condition (13)

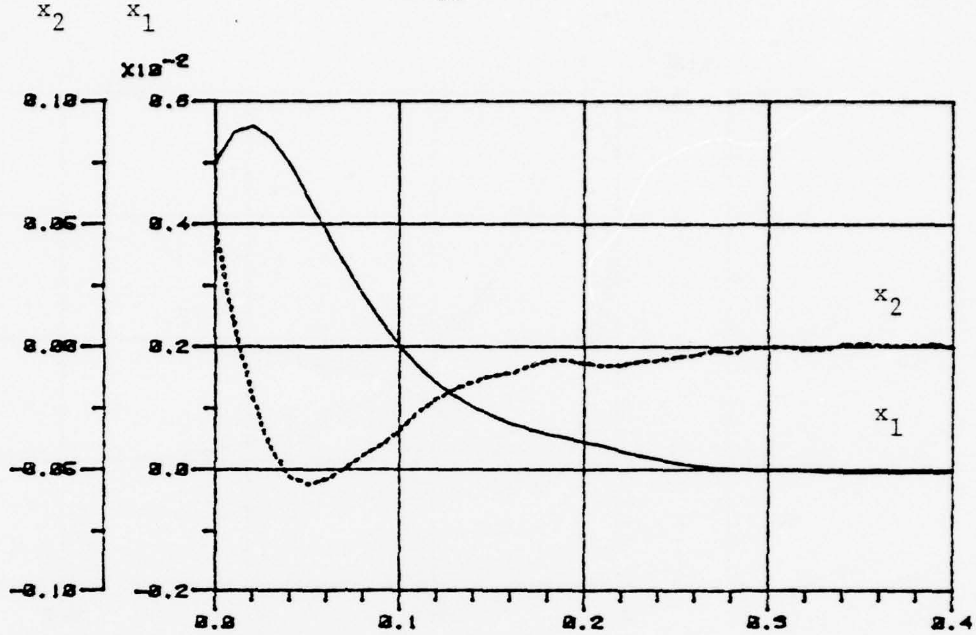


Figure 5.14a State response corresponding to condition (14)

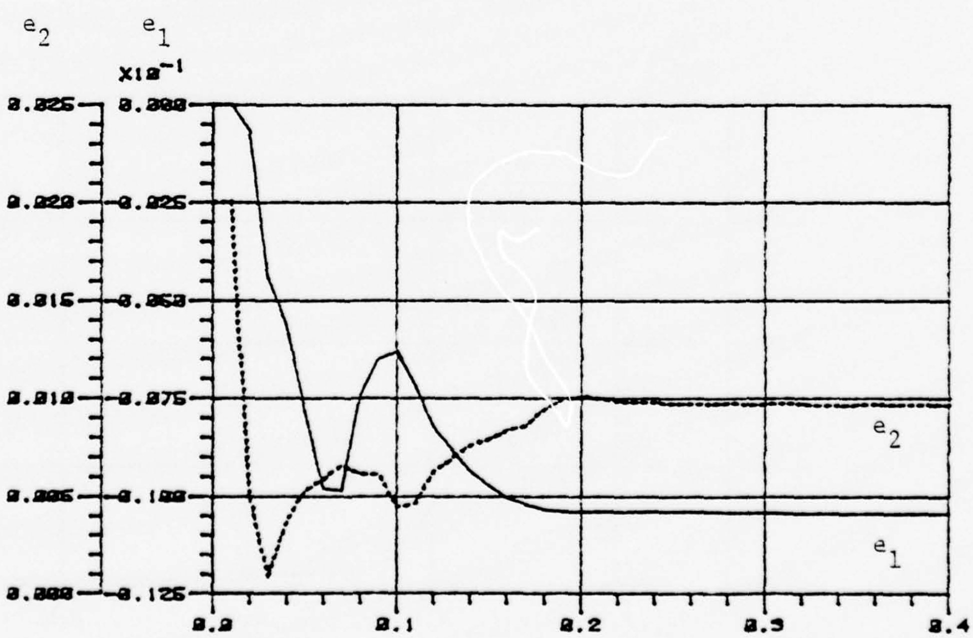


Figure 5.14b Time response of the estimates' error corresponding to condition (14)

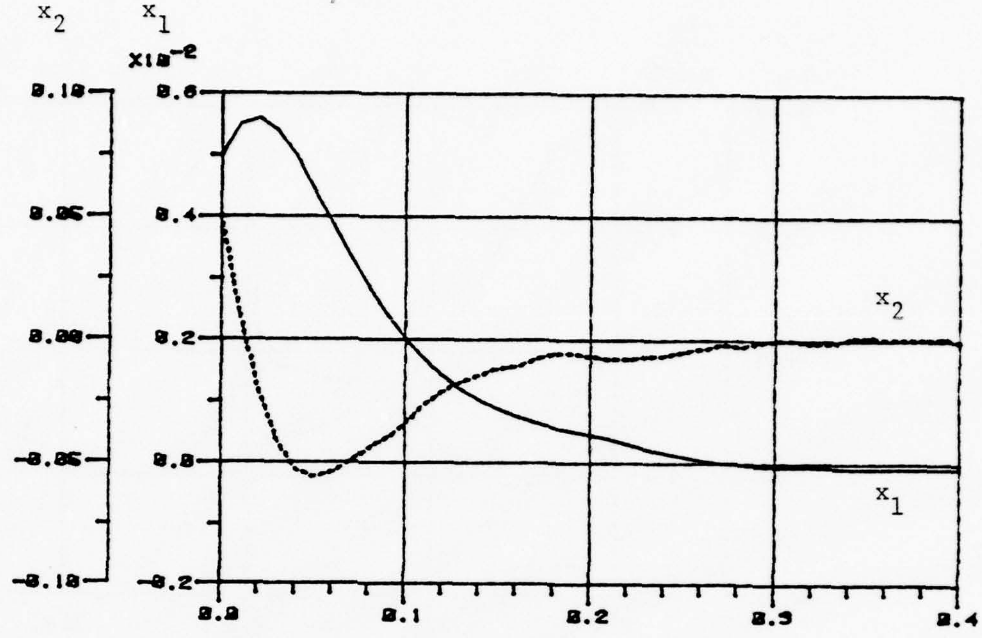


Figure 5.15a State response corresponding to condition (15)

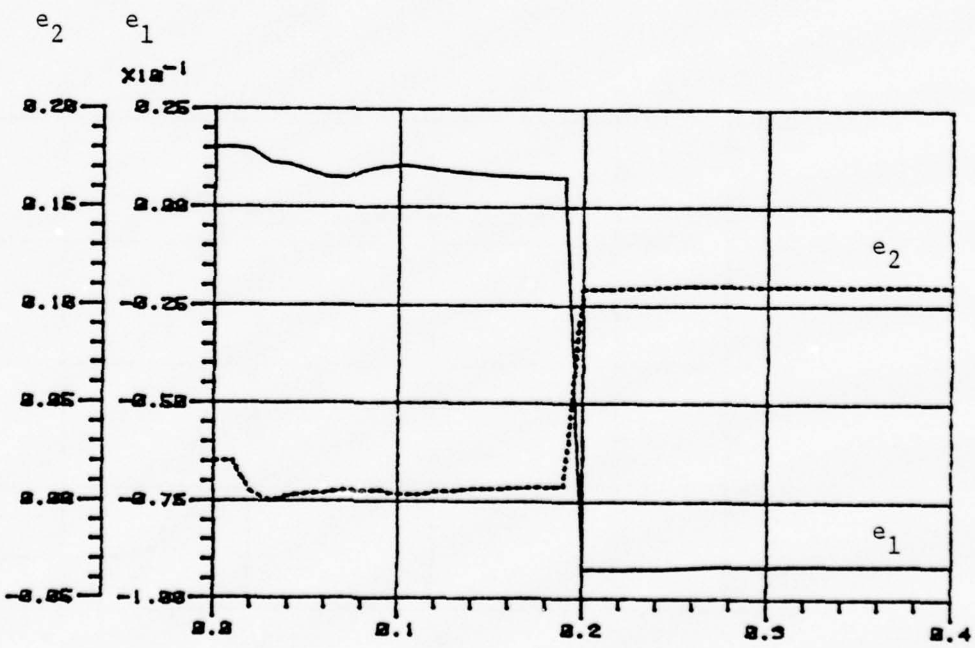


Figure 5.15b Time response of the estimates' error corresponding to condition (15)

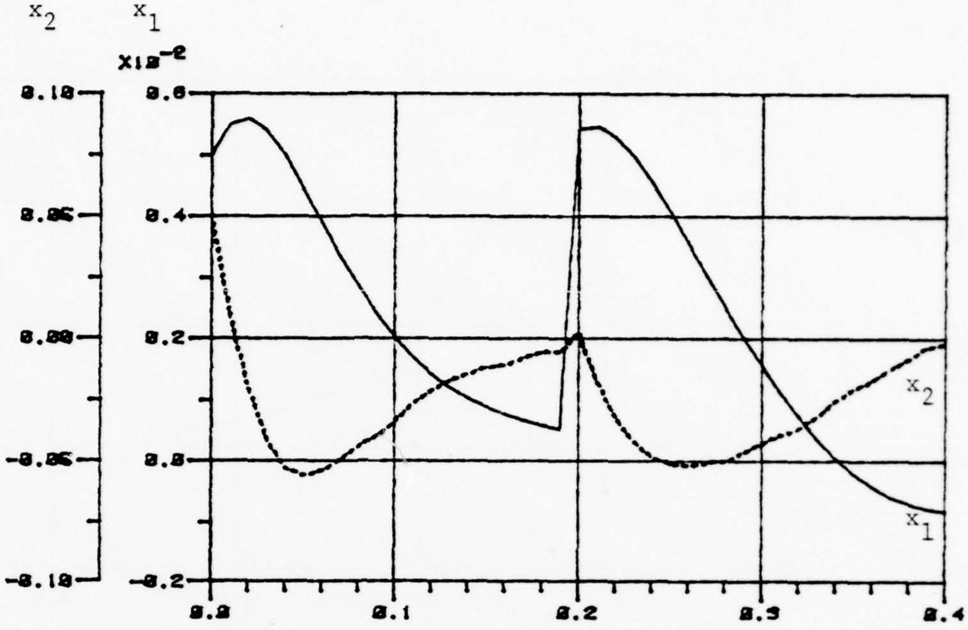


Figure 5.16a State response corresponding to condition (16)

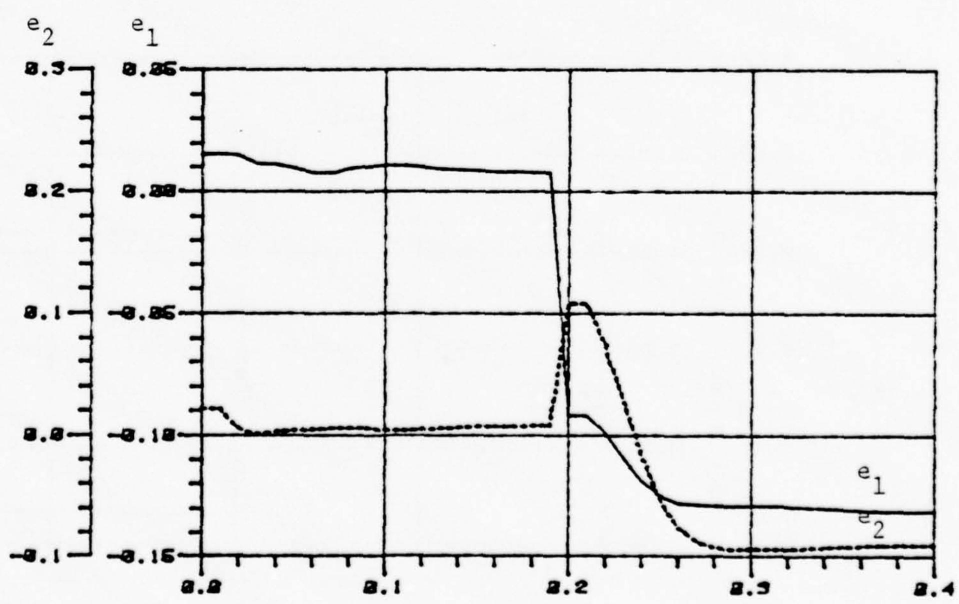


Figure 5.16b Time response of the estimates' error corresponding to condition (16)

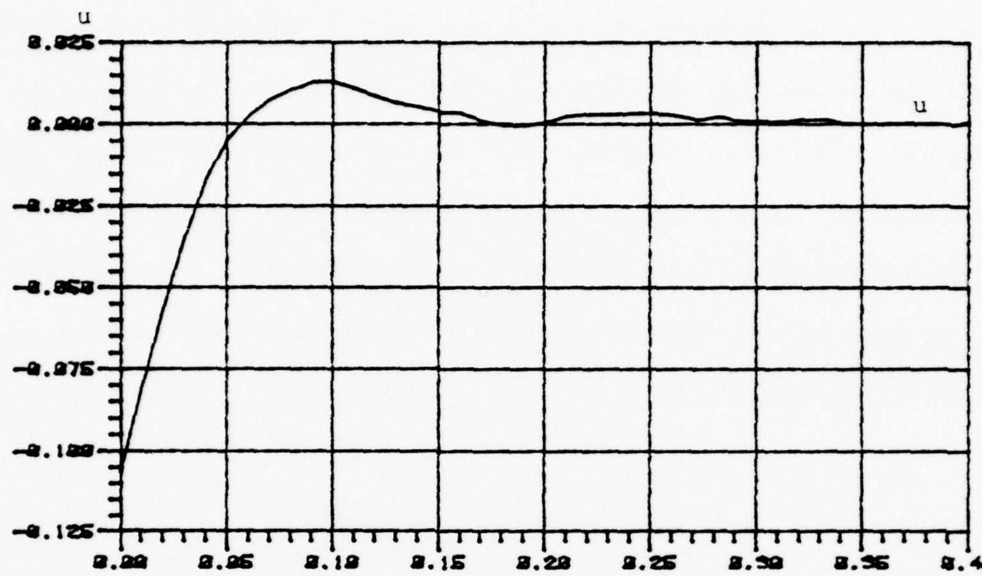


Figure 5.15c Control response corresponding to condition (15)

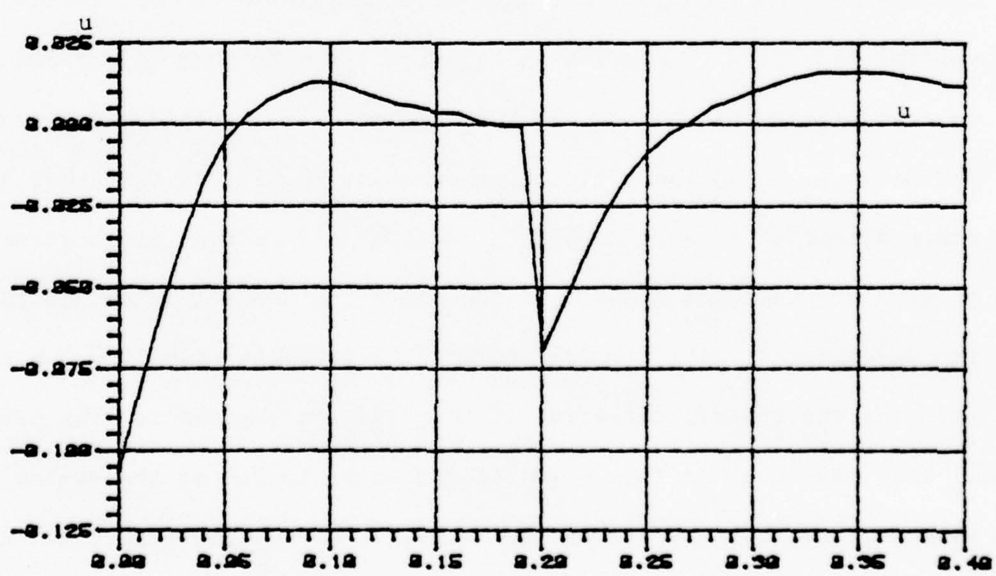


Figure 5.16c Control response corresponding to condition (16)

In condition (5), a milder position deviation, namely 0.005 m. is used as the initial condition. This yields a velocity response with smaller magnitude variation, which is desirable to the system. Hence, through the rest of the simulation, that is, for conditions (6-14), this particular initial position and velocity is used so that the resulting responses can be compared accordingly to extract some information on the characteristics of the system.

5.3.2. The Role of Control Signal in Estimation

As observed from the plots of the error of the parameters for conditions (1-5), the estimates for θ_2 are more accurate than those for θ_1. This is, in fact, a particularly good illustration of the feature of this SAFER control algorithm: the parameter which is more crucial to the control objective of the system is estimated with more effort and thus is more accurately estimated. In this magnetic ball suspension system, the parameter θ_1 is in the A matrix in equation (2.12); on the other hand, the other parameter θ_2 appears in the equation's B matrix, which directly governs the feedback gains, and thus also the control signal to the system. The parameter θ_2 will directly affect the system's stability, which constitutes the control objective of this system. Meanwhile, the parameter θ_1 does not have the same significance as θ_2 as far as the system's stability is concerned, thus its receiving a smaller estimation effort and not being estimated as accurately.

Another characteristic of this control algorithm that can be inferred from the response curves of the system is that as the states, thus also the feedback control signal, approach zero, the parameters are not estimated any more and steady state errors are incurred for them. One reason for this situation is that estimation requires part of the control signal. As the control signal approaches zero, even if the entire control signal strength is dedicated to estimation, there may still be insufficient effort for estimation. Another reason for this steady state error of the estimates is that as the system's states approach zero, the control objective of this system is essentially achieved and the need to estimate these unknown parameters to fulfill the control objective is not at issue any more. Incidentally, these points may serve as excellent illustrations to the statement that is made earlier in the thesis concerning the conflicting aspects of good control and good estimation: a large control signal may be needed for good estimation while a small signal may be required for good control.

5.3.3. Effect of Process Noise Level

The noise variance of w_k is set originally at 0.01. In conditions (6) and (7), this variance is changed to 0.5 and 0.0055 respectively, which represents a 50 fold increase and 0.055 times decrease respectively. This should be quite sufficient for obtaining some information on the characteristics of the system under various noise levels. The time res-responses corresponding to these conditions are shown in Figures 5.6a, 5.6b, 5.6c, 5.7a, 5.7b and 5.7c. Intuitively, a smaller noise variance should yield smoother responses, which is indeed the situation as shown in the response plots of Figures 5.7a and 5.7c for the states and control respectively. On the other hand, in Figures 5.6a and 5.6c, the states and control, which correspond to a much larger noise variance, do show more abrupt changes. Nonetheless, in both cases, stability of the system is achieved and the system is capable of operating under these noise levels satisfactorily.

From the response plots of the estimates' error in Figures 5.6b and 5.7b, the interplay between the control signal and estimation effort is again observed. In Fig. 5.6b, the parameters, especially θ_2 which influences the control signal, exhibit a more oscillatory behavior, indicating a tendency for the system to estimate them as long as the strength of the control signal is not too weak.

5.3.4. Tracking Ability of Filter

To ensure that the system will operate in a satisfactory fashion even with the unknown parameters deviating quite a bit from the expected values, simulation is carried out to test the tracking capability of the estimator. To attain this goal, the values of the parameters in the plant, which are used to generate the true states or the measurement of the magnetic ball system, are varied individually as well as simultaneously by 20% to 25%. These conditions are tabulated in Table 3 as conditions (8) through (13). In all these cases, it can be inferred from their corresponding state response plots that as far as stability is concerned, the system is performing well even though steady state errors are incurred for the parameter estimates. The parameter θ_2 which directly affects the control signal is invariably tracked to within an error of magnitude 0.02, which is a

5% error. On the other hand, the estimate for θ_1 is comparatively worse off, with an error magnitude which runs to 0.13, which is a 26% error, in the worst case. This situation is probably due, again, to the significance of the parameter θ_2 to the control objective of the system, as explained in section 5.2.2.

From the response plots for conditions (8) through (13), it can also be concluded that the extended Kalman filter coupled with the SAFER Control Algorithm is indeed tracking the parameter which is significant to the control objective of the system. Another characteristic concerning the system that may be inferred from these simulations is that the system is relatively insensitive to the parameter θ_1, thus tolerating a larger magnitude of error for it compared to the other parameter of the system. This point can be further illustrated from the response plots for condition (14), Figures 5.14a and 5.14b. In condition (14), the initial estimate for θ_1 is set at the value used in the plant. From Fig. 5.14b, it is observed that even with zero initial error, the estimate for θ_1 settles at another value. However, as in earlier cases, stability of system is still achieved in spite of steady state errors in the estimates. This particular aspect of the system can certainly be tolerated if knowledge of the exact values of the unknown parameters is not strongly demanded.

5.3.5. Effect of Time Varying Parameters

In a real world magnetic ball suspension system, the mass of the ball, the bias current for the control system, and even the suspension structure may change due to environmental variations or the system's internal disturbances. Corrosion, for instance, may cause change in the mass of the ball and the suspension structure. Anyway, these changes will be reflected in the variation of the values for m, $\bar{i}$ and the coil structure constant c, which will in turn result in changes for the values of the parameters θ_1 and θ_2. To simulate these changes, the plant parameters are changed at $t = 0.2$ seconds in conditions (15) and (16) as tabulated in Table 3. Furthermore, in condition (16), in addition to the changes of plant parameters, the states (position and velocity) are perturbed so as to add a more realistic flavor to the simulation. The changing of the states at the instants when the parameters are varied can, for instance, model an increase in the mass of the ball, which in turn results in a further deviation from the nominal position and an increase in velocity of the ball.

From the response curves of condition (15) given in Figures 5.15a and 5.15b, it is observed that at the instants when the parameters are changed, the states have almost decayed to zero. Since the states have not been affected by the change, this together with the weakness of the feedback control signal results in a situation in which the system makes no attempt to estimate the new parameter values and simply tolerates the large steady state errors for the parameters. Reemphasizing the characteristic of this closed-loop system mentioned earlier: as the stability of the system is attained, the estimation of the parameters are not at issue any more.

In the simulation of condition (16), the perturbation of the states results in a stronger feedback control signal which will in turn allow for a larger estimation effort. Furthermore, since the system's stability is disturbed during this change of the parameters, more effort is put into the estimation task to estimate the parameter which is crucial to the fulfillment of the system's objective. Indeed, from the plot of the error of the estimates in Fig. 5.16b, it is observed that after the parameters are changed and the states perturbed, the filter exhibits an effort to track the parameters. However, the estimates finally settle with some steady state error as stability of the system is regained and accurate estimation of the parameters is no longer necessary.

CHAPTER 6

CONCLUSION

In this thesis, application of the Sensitivity Adaptive Feedback with Estimation Redistribution (SAFER) control algorithm to the magnetic ball suspension system is considered. This particular study serves as an example where an adaptive control algorithm is applied to a stochastic system with unknown parameters.

From the simulation results, it can be concluded that the magnetic ball suspension system can indeed operate satisfactorily under various levels of noise disturbances and rather large deviation from nominal position under the application of a sensitivity approach adaptive control algorithm. It should be noted that the criterion for performance of the system is primarily its stability rather than the accurate estimation of the unknown parameters in the system. The feature of this sensitivity approach control algorithm is its capability of distributing more estimation effort for estimation of the parameters which is more significant to the control objective of the system. This is well exhibited in the simulation work done in this thesis.

Factorization techniques are applied to various matrix manipulation algorithms which are needed in the controller and estimator design. The utilization of these factorization algorithms has enhanced numerical stability and in particular, eliminated numerical underflow problems for the computations in the extended Kalman filter designed.

Meanwhile, there does exist hindrance to the application of this sophisticated and rather complex control algorithm to control systems in real time. This control algorithm may be more applicable to processes which advance relatively slow in time, certain chemical processes, for instance. Alternatives to overcome the lengthy computational time may include further simplification of the control algorithm, or reducing the dimensionality of the model as well as search for more efficient computational methods and devices.

REFERENCES

1. Bertsekas, Dimitri P., Dynamic Programming and Stochastic Control, Academic Press, New York, 1976.

2. Bierman, Gerald J., Factorization Methods for Discrete Sequential Estimation, Academic Press, New York, 1977.

3. Bierman, G. J. and Thornton, C. L., "Filtering and Error Analysis via the UDU^T Covariance Factorization", IEEE Trans. on Automatic Control, Vol. AC-23, No. 5, pp. 901-907, Oct. 1978.

4. Bryson, Arthur E. Jr., "Kalman Filter Divergence and Aircraft Motion Estimators", J. Guidance and Control, Vol. 1, No. 1, pp. 71-79, Jan-Feb 1978.

5. Chrisinger, J. E. et al., "Magnetic Suspension and Balance System for Wind Tunnel Application", J. Roy. Aeron. Soc., 67, pp. 717-724, 1963.

6. Cruz, J. B., Jr., Feedback Systems, McGraw-Hill, New York, 1972.

7. Gelb, Arthur, Applied Optimal Estimation, The M.I.T. Press, Cambridge, 1974.

8. Jacobson, Robert A. and Thornton, Catherine L., "Linear Stochastic Control Using the UDU^T Matrix Factorization," J. Guidance and Control, Vol. 1, No. 4, pp. 232-236, July 1978.

9. Jayawant, B. V. et al., "Development of 1-ton magnetically suspended vehicle using controlled d.c. electromagnets", Proc. IEE, Vol. 123, No. 9, pp. 941-948, Sept. 1976.

10. Melcher, James R. and Woodson, Herbert H., Electromechanical Dynamics: Part I: Discrete Systems, John Wiley & Sons, Inc., New York, 1968.

11. Mendel, Jerry M., Discrete Techniques of Parameter Estimation: The Equation Error Formulation, Marcel Dekker, Inc., New York, 1973.

12. Padilla, Consuelo S., "Stochastic Control for Systems with Uncertain Parameters", Ph.D. thesis, University of Illinois, July 1976, CSL Report R-734.

13. Padilla, C. S. and Cruz, J. B., Jr., "Sensitivity Adaptive Feedback with Estimation Redistribution", IEEE Trans. on Automatic Control, Vol. AC-23, No. 3, pp. 445-451, June 1978.

14. Padilla, C. S. and Cruz, J. B., Jr., "Output Feedback SAFER Control", Proc. International Conf. on Cybernetics and Society, Tokyo, Japan, pp. 1076-1080, November 1978.

15. Saridis, George N., Self-Organizing Control of Stochastic Systems, Marcel Dekker, Inc., New York, 1977.

16. Wittenmark, B., "Stochastic Adaptive Control Methods: A Survey", Int. J. Control, Vol. 21, No. 5, pp. 705-730, 1973.

APPENDIX A

DERIVATION OF THE COST EQUATIONS

This appendix is concerned with the derivation of the cost equations (3.14) and (3.15). The objective is to find the cost J given by equation (3.9). Identical results can be found in reference [12].

Given the cost J

$$J = E\{\sum_{k=N_o}^{N_o+\nu-1} \xi_k' \tilde{Q}_k \xi_k + u_k' R u_k\} \quad \text{(A.1)}$$

Assume the cost at $N_o+\nu$, $J_{N_o+\nu} = 0$, then the cost to go at $N_o+\nu-1$ can be written as:

$$J_{N_o+\nu-1} = E\{\xi'_{N_o+\nu-1}\bar{P}_{N_o+\nu-1}\xi_{N_o+\nu-1}\} + \alpha_{N_o+\nu-1} \quad \text{(A.2)}$$

where

$$\bar{P}_{N_o+\nu-1} = \tilde{Q}_{N_o+\nu-1}$$

$$\alpha_{N_o+\nu-1} = \alpha_{N_o+\nu} + E\{(\Gamma W_{N_o+\nu-1})'\bar{P}_{N_o+\nu-1}(\Gamma W_{N_o+\nu-1})\} \quad \text{(A.3)}$$

$$\alpha_{N_o+\nu} = 0$$

Continuing in this fashion, the cost to go at stage k can be written as:

$$J_k = E\{\sum_{i=k}^{N_o+\nu-1} \xi_i'(\tilde{Q}_i + K_i'RK_i)\xi_i\}$$

$$= E\{\xi_k'(\tilde{Q}_k + K_k'RK_k)\xi_k\} + J_{k+1} \quad \text{(A.4)}$$

Assuming the cost to go J_k to be of the form:

$$J_k = E\{\xi_k' \bar{P}_k \xi_k\} + \alpha_k \tag{A.5}$$

Substitute equation (A.5) into equation (A.4):

$$J_k = E\{\xi_k'(\tilde{Q}_k + K_k' R K_k)\xi_k\} + E\{\xi_{k+1}' \bar{P}_{k+1} \xi_{k+1}\} + \alpha_{k+1} \tag{A.6}$$

Using the system equation (3.8) with the matrices $\tilde{A}_k^{N_o}$, $\tilde{B}_k^{N_o}$ replacing $\tilde{A}_k$, $\tilde{B}_k$ to obtain ξ_{k+1} in equation (A.6), we have:

$$\begin{aligned} J_k = {} & E\{\xi_k'(\tilde{Q}_k + K_k' R K_k)\xi_k\} \\ & + E\{\xi_k'(\tilde{A}_k^{N_o}(K_k) - \tilde{B}_k^{N_o} K_k)' \bar{P}_{k+1} (\tilde{A}_k^{N_o}(K_k) - \tilde{B}_k^{N_o} K_k)\xi_k\} \\ & + E\{(\Gamma_{w_k})' \bar{P}_{k+1} (\Gamma_{w_k})\} + \alpha_{k+1} \end{aligned} \tag{A.7}$$

or

$$\begin{aligned} J_k = {} & E\{\xi_k'[\tilde{Q}_k + K_k' R K_k + \\ & (\tilde{A}_k^{N_o}(K_k) - \tilde{B}_k^{N_o} K_k)' \bar{P}_{k+1} (\tilde{A}_k^{N_o}(K_k) - \tilde{B}_k^{N_o} K_k)]\xi_k\} \\ & + (V_w)\mathrm{trace}\{\Gamma' \bar{P}_{k+1} \Gamma\} + \alpha_{k+1} \end{aligned}$$

Comparing (A.5) with (A.7), we have the following:

$$\bar{P}_k = \tilde{Q}_k + K_k' R K_k + (\tilde{A}_k^{N_o}(K_k) - \tilde{B}_k^{N_o} K_k)\bar{P}_{k+1}(\tilde{A}_k^{N_o}(K_k) - \tilde{B}_k^{N_o} K_k)$$

$$\bar{P}_{N_o+\nu-1} = \tilde{Q}_{N_o+\nu-1} \qquad k = N_o, \ldots, N_o+\nu-2 \tag{A.8}$$

$$\alpha_k = \alpha_{k+1} + V_w \ \mathrm{trace}\{\Gamma' \bar{P}_{k+1} \Gamma\} \tag{A.9}$$

$$\alpha_{N_o+\nu-1} = 0 \qquad k = N_o, \ldots, N_o+\nu-2$$

Combining equations (A.4), (A.7), (A.8) and (A.9), the cost at N_o can be written as:

$$J_{N_o} = \xi'_{N_o} \bar{P}_{N_o} \xi_{N_o} + (V_w) \sum_{i=N_o+1}^{N_o+\nu-1} \text{trace}\{\Gamma' \bar{P}_i \Gamma\}$$

$$= \xi'_{N_o} \bar{P}_{N_o} \xi_{N_o} + (V_w) \text{trace}\{\Gamma' (\sum_{i=N_o+1}^{N_o+\nu-1} \bar{P}_i) \Gamma\}$$

A final note is that this optimal cost J_{N_o} can also be expressed as:

$$J_{N_o} = \xi'_{N_o} \bar{P}_{N_o} \xi_{N_o} + (V_w) \text{trace}\{(\sum_{i=N_o+1}^{N_o+\nu-1} \bar{P}_i) \Gamma \Gamma'\}$$

since trace{AB} = trace{BA} assuming A and B are matrices of conformable dimension.

APPENDIX B

LIST OF VALUES FOR MATRICES IN EQUATIONS (3.12) AND (3.13)

The values of the matrices in equations (3.12) and (3.13) are:

$$\hat{A}_{N_o} = \begin{bmatrix} 1 & 0.01 \\ \hat{\theta}_1(N_o) & 1 \end{bmatrix} \qquad \hat{B}_{N_o} = \begin{bmatrix} 0 \\ \hat{\theta}_2(N_o) \end{bmatrix}$$

$$A_{\theta_1} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \qquad B_{\theta_1} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$A_{\theta_2} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \qquad B_{\theta_2} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$R = 0.5$

$\lambda = 1$

$$\tilde{Q} = \begin{bmatrix} Q & 0 \\ 0 & \lambda W^{N_o} \end{bmatrix} \qquad Q = \begin{bmatrix} 400 & 0 \\ 0 & 0.2 \end{bmatrix}$$

Corners of the weighting matrix W^{N_o} are:

(i) $W^{N_o} = \text{diag}\,(0.7, 0.1, 0.1, 0.1)$

(ii) $W^{N_o} = \text{diag}\,(0.1, 0.7, 0.1, 0.1)$

(iii) $W^{N_o} = \text{diag}\,(0.1, 0.1, 0.7, 0.1)$

(iv) $W^{N_o} = \text{diag}\,(0.1, 0.1, 0.1, 0.7)$

APPENDIX C

FILTER FACTORIZATION ALGORITHM AND RANK-ONE UPDATE ALGORITHM

Problem Formulation for Filter Factorization

The filter factorization and rank-one update algorithms, which can be found in references [2,3], are summarized below:

Given the linear dynamic system

$$x_{i+1} = \phi_i x_i + \Gamma_i u_i + G_i w_i \tag{C.1}$$

and observation

$$z_i = H_i x_i + v_i \tag{C.2}$$

where $x_i \in R^n$, $z_i \in R^m$, w_i, v_i are zero mean and x_o, w_i, v_i are statistically independent Gaussian variables with covariances

$$E\{w_i w_j^T\} = Q_i \delta_{ij}, \quad E\{v_i v_j^T\} = R_i \delta_{ij}, \quad E\{x_o x_o^T\} = \bar{P}_o$$

The vector $\hat{x}_i$ represents the minimum variance estimate of x_i given z_o, $z_1, \ldots, z_i$ and may be obtained from the following Kalman filter algorithm.

Measurement update:

$$\hat{x}_i = \tilde{x}_i + K_i(z_i - H_i \tilde{x}_i) \tag{C.3}$$

$$K_i = \bar{P}_i H_i^T (H_i \bar{P}_i H_i^T + R_i)^{-1} \tag{C.4}$$

$$P_i = \bar{P}_i - K_i H_i \bar{P}_i \tag{C.5}$$

Time update:

$$\tilde{x}_{i+1} = \Phi_i \hat{x}_i + \Gamma_i \hat{u}_i \qquad (C.6)$$

$$\bar{P}_{i+1} = \Phi_i P_i \Phi_i^T + G_i Q_i G_i^T \qquad (C.7)$$

where $\ddot{u}_i$ is the optimal control value.

Initial conditions

$$\tilde{x}_o = E\{x_o\} \quad \text{given} \qquad (C.8)$$

$$\bar{P}_o \quad \text{given} \qquad (C.9)$$

Suppose the n-dimensional error covariance matrix P, is factored such that

$$P = UDU^T \qquad (C.10)$$

where U is upper triangular with unit diagonals and $D = \text{diag}(d_1, \ldots, d_n)$. The matrices U and D are referred to as the U-D factors of P. They are unique, provided that P is positive definite and can be constructed using a Cholesky factorization [2].

Algorithms are presented for performing measurement and time updating of the U-D factors. These algorithms correspond to the conventional Kalman formulas of equations (C.4), (C.5) and (C.7).

U-D Measurement Update Algorithm

Given a priori covariance factors U and D and a scalar measurement $z = Hx + v$, where $E\{v^2\} = r$, the updated U-D covariance factors and the Kalman gain (U, D and K respectively) can be obtained as follows:

$$f^T = H\bar{u} \qquad f^T = (f_1, \ldots, f_n) \tag{C.11}$$

$$v = \bar{D}f \qquad v_j = \bar{d}_j f_j \tag{C.12}$$

$$\bar{K}_1^T = (v_1, \overbrace{0, \ldots, 0}^{n-1}) \tag{C.13}$$

$$\alpha_1 = r + v_1 f_1 \tag{C.14}$$

If $\alpha_1 = 0$, omit equation (C.15)

$$d_1 = (r/\alpha_1)\bar{d}_1 \tag{C.15}$$

For $j = 2, \ldots, n$ cycle through equations (C.16-C.20)

$$\alpha_j = \alpha_{j-1} + v_j f_j \tag{C.16}$$

If $\alpha_j = 0$, omit equations (C.17-C.20)

$$d_j = (\alpha_{j-1}/\alpha_j)\bar{d}_j \tag{C.17}$$

If $d_j = 0$, skip to equation (C.20)

$$\lambda_j = f_j/\alpha_{j-1} \tag{C.18}$$

$$U_j = \bar{U}_j - \lambda_j \bar{K}_{j-1} \tag{C.19}$$

$$\bar{K}_j = \bar{K}_{j-1} + v_j \bar{U}_j \tag{C.20}$$

where $U = [U_1, \ldots, U_n]$. The component U vectors have the form

$$U_j^T = [U_j(1), \ldots, U_j(j-1), 1, 0, \ldots, 0]$$

and $D = \text{diag}(d_1, \ldots, d_n)$. The Kalman gain is given by

$$K = \bar{K}_n / \alpha_n \qquad \text{(C.21)}$$

where α_n is the innovation covariance.

Modified Gram-Schmidt Time Update Algorithm

Modified Gram-Schmidt techniques may be used to accomplish time updating of the U-D factors, and the resulting algorithm is the following. Let

$$W = [\overbrace{\Phi U}^{n} \quad \overbrace{G}^{p}] \equiv \begin{bmatrix} w_1^T \\ w_2^T \\ \vdots \\ w_n^T \end{bmatrix}$$

$$\bar{D} = \text{diag}(\overbrace{D,}^{n} \overbrace{Q}^{p}) = \text{diag}(\bar{d}_1, \bar{d}_2, \ldots, \bar{d}_{n+p})$$

The $\tilde{U}$-$\tilde{D}$ factors of $\bar{P} = W\bar{D}W^T$ in equation (C.7) may be computed as follows: For $j = n, n-1, \ldots, 1$ evaluate equations (C.22-C.24)

$$\tilde{d}_j = w_j^T \bar{D} w_j \qquad \text{(C.22)}$$

If $d_j = 0$, omit equations (C.23-C.24)

$$\tilde{u}_{ij} = \frac{1}{\bar{d}_j} [w_i^T \bar{D} w_j] \tag{C.23}$$

$$w_i := w_i - \tilde{u}_{ij} w_j \tag{C.24}$$

where the symbol ":=" denotes replacement in computer storage.

Rank-One Update Algorithm

Let

$$\bar{P} = \overline{UDU}^T = UDU^T + caa^T$$

where c is a scalar, a is an n vector, U is unit upper triangular, $D = \mathrm{diag}(d_1, \ldots, d_n)$ and $n = \dim P$.

If $\bar{P}$ is positive definite, then the factors $\bar{U}$ and $\bar{D}$ can be calculated as follows:

For $j = n, n-1, \ldots, 2$ recursively evaluate ordered equations (C.25) through (C.28):

$$\bar{d}_j = d_j + c_j a_j^2 \qquad (c_n = c) \tag{C.25}$$

$$\left.\begin{aligned} a_k &:= a_k - a_j U(k,j) && \text{(C.26)} \\ \bar{U}(k,j) &= U(k,j) + c_j a_j a_k / \bar{d}_j && \text{(C.27)} \end{aligned}\right\} k = 1, \ldots, j-1$$

$$c_{j-1} = c_j d_j / \bar{d}_j \tag{C.28}$$

and then compute

$$\bar{d}_1 = d_1 + c_1 a_1^2 .$$

APPENDIX D

LISTING OF COMPUTER PROGRAMS

```
C TO EXECUTE THIS PROGRAM IN DEC-10 COMPUTER:
C
C (1) ASSIGN DSK:2 BEFORE EXECUTION
C (2) TO EXECUTE, TYPE THE FOLLOWING
C      EX FILENAME.EXTENSION, SYS:SSP10/SEARCH
C
C SUBROUTINES FOR MATRIX MANIPULATIONS
C THE MAXIMUM DIMENSION OF MATRIX IS (20,20)
C
C
C
C
C SCALAR MULT. OF MATRIX
C A IS OF DIMENSION I*J, S IS THE SCALAR
      SUBROUTINE MATSM(A,S,I,J)
      REAL A(I,J)
      DO 20 M=1,I
      DO 20 N=1,J
20    A(M,N)=A(M,N)*S
      RETURN
      END
C
C
C
C  MATRIX MULTIPLICATION
C A IS I*J, B IS J*K, RESULT IS C (I*K)
      SUBROUTINE MATMUL(A,B,C,I,J,K)
      REAL A(I,J), B(J,K), C(I,K)
      REAL E(20,20)
      DO 20 M=1,I
      DO 20 N=1,K
      E(M,N)=0
      DO 20 L=1,J
20    E(M,N)=E(M,N)+A(M,L)*B(L,N)
      DO 30 M=1,I
      DO 30 N=1,K
30    C(M,N)=E(M,N)
      RETURN
      END
```

```
C
C
C
C MATRIX TRANSPOSE
C A IS INPUT MATRIX OF I*J
        SUBROUTINE MATTNS(A,AT,I,J)
        REAL A(I,J), AT(J,I)
        REAL AE(20,20)
        DO 20 M=1,J
        DO 20 N=1,I
20      AE(M,N)=A(N,M)
        DO 30 M=1,J
        DO 30 N=1,I
30      AT(M,N)=AE(M,N)
        RETURN
        END
C
C
C
C MATRIX ADDITION(1)/SUBTRACTION(2)
C A, B ARE OF DIMENSION I*J, C IS RESULT
C L IS 1 FOR ADDITION, 2 FOR SUBRACTION (A-B)
        SUBROUTINE MATAS(A,B,C,I,J,L)
        REAL A(I,J), B(I,J), C(I,J)
        IF(L.EQ.2) GOTO 40
        DO 20 M=1,I
        DO 20 N=1,J
20      C(M,N)=A(M,N)+B(M,N)
        RETURN
40      DO 50 M=1,I
        DO 50 N=1,J
50      C(M,N)=A(M,N)-B(M,N)
        RETURN
        END
C
C
C
C CALCULATE TRACE OF MATRIX
C A IS OF DIMENSION L*L, S IS RESULT
        SUBROUTINE TRACE(A,S,L)
        REAL A(L,L)
        S=0
        DO 20 N=1,L
20      S=S+A(N,N)
        RETURN
        END
C
C
C
C CALCULATE EUCLIDEAN NORM OF A VECTOR
C A IS N*1 VECTOR, XN IS RESULTING NORM VALUE
        SUBROUTINE NORM(A,N,XN)
        DIMENSION A(N,1)
        XN=0
        DO 10 I=1,N
```

```
10      XN=XN+A(I,1)*A(I,1)
        XN=SQRT(XN)
        RETURN
        END
C
C
C
C SUBROUTINE FOR INITIALIZING AN IDENTITY MATRIX
C A IS TO SET TO IDENTITY MATRIX OF N*N
        SUBROUTINE MATID(A,N)
        REAL A(N,N)
        CALL MATSM(A,0.,N,N)
        DO 10 I=1,N
10      A(I,I)=1.
        RETURN
        END
C
C
C
C SUBROUTINE FOR INITIALIZING AN IDENTITY MATRIX
C A IS TO SET TO IDENTITY MATRIX OF N*N
        SUBROUTINE MATID(A,N)
        REAL A(N,N)
        CALL MATSM(A,0.,N,N)
        DO 10 I=1,N
10      A(I,I)=1.
        RETURN
        END
C
C
C
C
C
C
C THIS SUBROUTINE GENERATES A SEQUENCE OF INDEPENDENT
C GAUSSIAN RANDOM NUMBERS WITH 'MEAN' AND 'VAR', WHEN
C CALLED SUCCESSIVELY. 'MEAN' AND 'VAR' CAN BE VARIED
C EACH TIME THE SUBROUTINE IS CALLED.
C THE FIRST TIME THE SUBROUTINE IS CALLED WITHIN A MAIN
C PROGRAM THE FIRST ARGUMENT 'I' SHOULD BE EQUAL TO
C UNITY.
C THE MAIN PROGRAM SHOULD BE EXECUTED
C CALLING SYS:SSP10/SEARCH
C
        SUBROUTINE RANDOM(I,VAR,MEAN,RV,KKK)
        REAL MEAN
        IF(I.EQ.1) KKK=1
        CALL SETRAN(KKK)
        Y=RAN(1)
        CALL NDTRI(Y,X,Z,IER)
        RV=MEAN+X*SQRT(VAR)
        CALL SAVRAN(KKK)
        RETURN
        END
C
```

```
C
C
C
C
C
C SUBROUTINE FOR WRITING MATRIX ON DISK FOR LAS ANALYSIS
C
C A IS M*N MATRIX
C
C MATRIX NAMES SHOULD BE GENERATED IN THE FILE
C NAMES.DAT SUCH THAT EACH MATRIX RECORD WOULD
C CORRESPOND TO A PARTICULAR NAME
C
C ALSO, PERFORM "ASSIGN DSK:2" BEFORE MAIN PROGRAM
C EXECUTION
C
        SUBROUTINE WDSK(A,M,N,NREC)
        DIMENSION A(M,N)
        DIMENSION RECOR(103)
        CALL DEFINE FILE(2,103,NV,'MATRIC.DAT',0,0)
        RECOR(1)=M*N
        RECOR(2)=M
        RECOR(3)=N
        NCON=0
        DO 10 I=1,N
        DO 20 J=1,M
20      RECOR(3+J+NCON)=A(J,I)
10      NCON=I*M
        WRITE(2 NREC)RECOR
        TYPE 111, NREC
111     FORMAT(/'   MATRIX HAS RECORD    = ',I6)
        RETURN
        END
C
C
C
C
C
C
C SUBROUTINES FOR GENERATING NEW STATES WITH
C LINEARIZED DISCRETE PLANT OF EQN. (3.8)
C
C
C
C THIS SUBROUTINE INITIALIZE THE SYSTEM EQUATION
C A MATRIX WITH PLANT PARAMETER VALUES
C
        SUBROUTINE SETRA(A)
        COMMON A1,A2,A3,A4,B2,RA3,RB2
        REAL A(6,6)
        DO 10 M=1,6
        DO 10 N=1,6
10      A(M,N)=0
        A(1,1)=A1
        A(1,2)=A2
```

```
      A(2,1)=RA3
      A(2,2)=A4
      A(3,3)=A1
      A(3,4)=A2
      A(4,1)=1
      A(4,3)=RA3
      A(4,4)=A4
      A(5,5)=A1
      A(5,6)=A2
      A(6,5)=RA3
      A(6,6)=A4
      RETURN
      END
C
C
C
C CALCULATE SYSTEM MATRIX GIVEN FEEDBACK GAINS
C
      SUBROUTINE CALRA(A,K,IN)
      COMMON A1,A2,A3,A4,B2,RA3,RB2
      REAL A(6,6), K(6,20)
      A(4,3)=RA3-RB2*K(1,IN)
      A(4,4)=A4-RB2*K(2,IN)
      DO 10 N=1,4
10    A(6,N)=-K(N,IN)
      A(6,5)=RA3-RB2*K(1,IN)-K(5,IN)
      A(6,6)=A4-RB2*K(2,IN)-K(6,IN)
      RETURN
      END
C
C
C
C CALCULATE MATRIX FOR COVARIANCE PROPAGATION
      SUBROUTINE CALRB(A,K,IN)
      COMMON A1,A2,A3,A4,B2,RA3,RB2
      REAL A(6,6),K(6,20)
      A(2,1)=RA3-RB2*K(1,IN)
      A(2,2)=A4-RB2*K(2,IN)
      DO 20 N=3,6
20    A(2,N)=-RB2*K(N,IN)
      RETURN
      END
C
C
C
C THE FOLLOWING ARE SUBROUTINES FOR THE
C MAGNETIC BALL SUSPENSION MODEL
C
C THE ESTIMATES FOR THE PARAMETERS ARE
C USED THROUGHOUT THESE COMPUTATIONS
C
C
C SUBROUTINE TO INITIALIZE SYSTEM MATRIX A
      SUBROUTINE SETA(A)
      COMMON A1,A2,A3,A4,B2,RA3,RB2
```

```
      DIMENSION A(6,6)
      DO 10 M=1,6
      DO 10 N=1,6
10    A(M,N)=0
      A(1,1)=A1
      A(1,2)=A2
      A(2,1)=A3
      A(2,2)=A4
      A(3,3)=A1
      A(3,4)=A2
      A(4,1)=1
      A(4,3)=A3
      A(4,4)=A4
      A(5,5)=A1
      A(5,6)=A2
      A(6,5)=A3
      A(6,6)=A4
      RETURN
      END
C
C
C
C CALCULATE SYSTEM MATRIX GIVEN FEEDBACK GAINS
      SUBROUTINE CALA(A,K,IN)
      COMMON A1,A2,A3,A4,B2,RA3,RB2
      REAL A(6,6), K(6,20)
      A(4,3)=A3-B2*K(1,IN)
      A(4,4)=A4-B2*K(2,IN)
      DO 10 N=1,4
10    A(6,N)=-K(N,IN)
      A(6,5)=A3-B2*K(1,IN)-K(5,IN)
      A(6,6)=A4-B2*K(2,IN)-K(6,IN)
      RETURN
      END
C
C
C
C CALCULATE MATRIX FOR COVARIANCE PROPAGATION
      SUBROUTINE CALB(A,K,IN)
      COMMON A1,A2,A3,A4,B2,RA3,RB2
      REAL A(6,6),K(6,20)
      A(2,1)=A3-B2*K(1,IN)
      A(2,2)=A4-B2*K(2,IN)
      DO 20 N=3,6
20    A(2,N)=-B2*K(N,IN)
      RETURN
      END
C
C
C
C SUBROUTINE TO CHANGE WEIGHTING MATRIX W
C VARIOUS CORNERS OF THE MATRIX W IS
C OBTAINED BY SPECIFYING THE VALUE
C OF THE INTEGER N (1,2,3,4).
C
```

```
C
C
        SUBROUTINE SWQD(QD,WNO,N,AL)
        REAL QD(6,6)
        RWNO=(1.-WNO)*AL/3.
        IF (N.NE.1) GOTO 20
        QD(3,3)=WNO*AL
        DO 10 I=4,6
10      QD(I,I)=RWNO
        RETURN
20      IF (N.NE.2) GO TO 40
        QD(4,4)=WNO*AL
        QD(3,3)=RWNO
        QD(5,5)=RWNO
        QD(6,6)=RWNO
        RETURN
40      IF (N.NE.3) GO TO 60
        QD(5,5)=WNO*AL
        QD(3,3)=RWNO
        QD(4,4)=RWNO
        QD(6,6)=RWNO
        RETURN
60      QD(6,6)=WNO*AL
        DO 70 I=3,5
70      QD(I,I)=RWNO
        RETURN
        END
C
C
C
C SUBROUTINE TO CALCULATE COST
C
C Z IS THE STATE VALUES AT TIME NO,
C A IS THE SYSTEM MATRIX, K IS THE SEQUENCE OF  FEEDBACK
C GAINS, QD IS THE AUGMENTED PENALTY MATRIX FOR THE
C STATES, GA IS THE AUGMENTED NOISE MATRIX, VW IS
C THE VARIANCE OF THE NOISE, MU IS THE NUMBER OF
C OPTIMIZATION STAGE, AND COST IS THE RESULT.
C
C
        SUBROUTINE COMPJ(Z,A,K,QD,GA,VW,COST,MU)
        REAL A(6,6), P(6,6), F(6,6), C(6,6), QD(6,6)
        REAL Z(6,1), D(6,1), E(1,6), AG(1,6), GA(6,1)
        REAL K(6,MU)
        DO 20 I=1,6
        DO 20 J=1,6
        F(I,J)=0
20      P(I,J)=QD(I,J)
        DO 11 M=MU-1,1,-1
        CALL MATAS(P,F,F,6,6,1)
        CALL SETA(A)
        MT=M
        CALL CALA(A,K,MT)
        CALL CALB(A,K,MT)
        CALL MATMUL(P,A,P,6,6,6)
```

```
      CALL MATTNS(A,C,6,6)
      CALL MATMUL(C,P,P,6,6,6)
      CALL MATAS(QD,P,P,6,6,1)
      DO 10 LL=1,6
10    E(1,LL)=K(LL,M)
      CALL MATTNS(E,D,1,6)
      CALL MATMUL(D,E,C,6,1,6)
      CALL MATAS(P,C,P,6,6,1)
11    CONTINUE
      CALL MATTNS(GA,AG,6,1)
      CALL MATMUL(AG,F,E,1,6,6)
      CALL MATMUL(E,GA,COST,1,6,1)
      COST=COST*VW
      CALL MATTNS(Z,E,6,1)
      CALL MATMUL(P,Z,D,6,6,1)
      CALL MATMUL(E,D,TT,1,6,1)
      COST=COST+TT
      RETURN
      END
C
C
C
C
C SUBROUTINE FOR COMPUTING FEEDBACK GAINS
C USING FACTORIZATION METHODS
C AND RANK ONE UPDATE ALGORITHM
C
C
C R IS THE WEIGHT FOR THE CONTROL IN THE COST EQUATION.
C
C
      SUBROUTINE COMKUD(A,B,QD,K,R,MU)
      REAL A(6,6), B(6,1), P(6,6), K(6,MU)
      REAL AT(6,6), QD(6,6)
      REAL V(6,1), U(6,6), UT(6,6), D(6,6), DT(6,6)
      REAL W(6,12), DB(12,12), WP(6,6), RWD(1,12)
      REAL TRWD(12,1)
      COMMON A1,A2,A3,A4,B2,RA3,RB2
C
C TO DETECT COMPUTATION OVERFLOW OR UNDERFLOW
      CALL ERRSET(100)
C
C
      CALL MATSM(W,0.,6,12)
      CALL MATSM(DB,0.,12,12)
      CALL MATSM(DT,0.,6,6)
C
      DO 10 I=1,6
      W(I,I+6)=1.
      DB(I+6,I+6)=QD(I,I)
      D(I,I)=QD(I,I)
10    CONTINUE
C
      CALL MATID(U,6)
C
```

```
C
C TO COMPUTE THE FEEDBACK CONTROL GAINS SEQUENCE
        DO 160 IN=MU-1,1,-1
C
        CALL MATTNS(U,UT,6,6)
        CALL MATMUL(D,UT,P,6,6,6)
        CALL MATMUL(U,P,P,6,6,6)
        SCA=R+P(2,2)*B2*B2
        SCB=SCA+B2*P(6,2)
        SCA=-1./SCA
        K(1,IN)=(A1*B2*P(1,2)+A3*B2*P(2,2)+B2*P(4,2))/SCB
        K(2,IN)=(A2*B2*P(1,2)+A4*B2*P(2,2))/SCB
        K(3,IN)=(A1*B2*P(3,2)+(A3-B2*K(1,IN))*B2*P(4,2))/SCB
        K(4,IN)=(A2*B2*P(3,2)+(A4-B2*K(2,IN))*B2*P(4,2))/SCB
        K(5,IN)=(A1*B2*P(5,2)+(A3-B2*K(1,IN))*B2*P(6,2))/SCB
        K(6,IN)=(A2*B2*P(5,2)+(A4-B2*K(2,IN))*B2*P(6,2))/SCB
        CALL SETA(A)
        INT=IN
        CALL CALA(A,K,INT)
        CALL MATTNS(A,AT,6,6)
C
C U-D FACTORIZATION METHOD
C
C
        CALL MATMUL(P,B,V,6,6,1)
        CALL MATMUL(AT,V,V,6,6,1)
        CALL MATMUL(AT,U,WP,6,6,6)
        DO 20 I=1,6
        DB(I,I)=D(I,I)
        DO 20 J=1,6
20      W(I,J)=WP(I,J)
C
        DO 80 JJ=6,1,-1
        DO 65 KL=1,12
65      RWD(1,KL)=W(JJ,KL)
        CALL MATTNS(RWD,TRWD,1,12)
        CALL MATMUL(DB,TRWD,TRWD,12,12,1)
        CALL MATMUL(RWD,TRWD,D(JJ,JJ),1,12,1)
        IF ( D(JJ,JJ).EQ.0.) GO TO 80
C
        DO 80 KK=1,JJ-1
        DO 85 LN=1,12
85      RWD(1,LN)=W(KK,LN)
C
        CALL MATMUL(RWD,TRWD,US,1,12,1)
        U(KK,JJ)=US/D(JJ,JJ)
C
        DO 95 LK=1,12
95      W(KK,LK)=W(KK,LK)-U(KK,JJ)*W(JJ,LK)
C
80      CONTINUE
C
C
        CALL MATTNS (U,UT,6,6)
        CALL MATMUL(D,UT,UT,6,6,6)
```

```
      CALL MATMUL(U,UT,UT,6,6,6)
C
C
C RANK ONE UPDATE
C
      CALL MATID(UT,6)
      DO 120 JK=6,2,-1
      DT(JK,JK)=D(JK,JK)+SCA*V(JK,1)*V(JK,1)
      DO 110 LK=1,JK-1
      V(LK,1)=V(LK,1)-V(JK,1)*U(LK,JK)
110   UT(LK,JK)=U(LK,JK)+SCA*V(JK,1)*V(LK,1)/DT(JK,JK)
      SCA=SCA*D(JK,JK)/DT(JK,JK)
120   CONTINUE
C
      DT(1,1)=D(1,1)+SCA*V(1,1)*V(1,1)
C
C
      DO 130 I=1,6
      D(I,I)=DT(I,I)
      DO 130 J=1,6
130   U(I,J)=UT(I,J)
C
C
      CALL MATTNS (U,UT,6,6)
      CALL MATMUL(D,UT,UT,6,6,6)
      CALL MATMUL(U,UT,UT,6,6,6)
C
160   CONTINUE
      RETURN
      END
C
C
C
C
C
C
C SUBROUTINE FOR MAGNETIC BALL SUSPENSION
C
C EXTENDED KALMAN FILTERING
C
C KF IS THE KALMAN FILTER GAINS, AUX IS THE 8-VECTOR
C FOR THE AUGMENTED EIGTH ORDER SYSTEM, Z IS THE
C MEASUREMENT VECTOR, K IS THE CONTROL FEEDBACK
C GAINS, R IS THE CONTROL WEIGHT IN THE COST
C EQUATION, H IS THE OBSERVATION MATRIX, VNW IS THE
C NOISE VARIANCE OF THE SYSTEM'S NOISE, AU IS THE
C 8*8 SYSTEM MATRIX, XN IS THE NOMINAL TRAJECTORY
C FOR THE KALMAN FILTER.
C
C
      SUBROUTINE KALMAN(KF,AUX,UT,DT,Z,K,R,H,VNW,AU,XN)
      REAL  AUX(8,1), XN(6,1)
      REAL U(8,8), D(8,8), UT(8,8), DT(8,8), Q(8,8)
      REAL RH(1,8), CKF(8,1)
      REAL KF(8,6), Z(6,1)
```

```
      REAL AU(8,8), A(6,6), K(6,20)
      REAL R(6,6), H(6,8)
      REAL F(8,1), DB(9,9), WD(8,9), RWD(1,9)
      REAL FT(1,8)
      REAL WP(8,8), TRWD(9,1)
      COMMON A1,A2,A3,A4,B2,RA3,RB2
C
      CALL ERRSET(100)
C
C
C INITIALIZE WD'S G COMPONENT
      CALL MATSM(WD,0.,8,9)
      WD(2,9)=A2
C
C
C INITIALIZE DB
      CALL MATSM(DB,0.,9,9)
      DB(9,9)=VNW
C
C INITIALIZE U, D
      CALL MATSM(U,0.,8,8)
      CALL MATSM(D,0.,8,8)
C
C ASSUME SCALAR MEASUREMENT, GO THRU. LOOP 6 TIMES
      DO 100 I=1,6
      RI=R(I,I)
      DO 11 JK=1,8
11    RH(1,JK)=H(I,JK)
      CALL MATMUL(RH,UT,FT,1,8,8)
      CALL MATTNS(FT,F,1,8)
      CALL MATSM(CKF,0.,8,1)
      AJS1=RI+DT(1,1)*F(1,1)*F(1,1)
      D(1,1)=RI*DT(1,1)/AJS1
      CKF(1,1)=DT(1,1)*F(1,1)
      U(1,1)=UT(1,1)
C
      DO 40 JK=2,8
      AJ=AJS1+DT(JK,JK)*F(JK,1)*F(JK,1)
      D(JK,JK)=AJS1/AJ*DT(JK,JK)
C     IF (D(JK,JK).EQ.0.) GO TO 37
      SL=F(JK,1)/AJS1
      DO 36 LL=1,8
36    U(LL,JK)=UT(LL,JK) - SL*CKF(LL,1)
C
C
39    DO 38 LL=1,8
38    CKF(LL,1)=CKF(LL,1)+DT(JK,JK)*F(JK,1)*UT(LL,JK)
C
40    AJS1=AJ
      AJ=1./AJ
      CALL MATSM(CKF,AJ,8,1)
      DO 50 JL=1,8
50    KF(JL,I)=CKF(JL,1)
      CALL MATMUL(RH,AUX,SM,1,8,1)
      SM=Z(I,1)-SM
```

```
        CALL MATSM(CKF,SM,8,1)
        CALL MATAS(AUX,CKF,AUX,8,1,1)
C
        DO 100 LOU=1,8
        DO 100 LIN=1,8
        UT(LOU,LIN)=U(LOU,LIN)
        DT(LOU,LIN)=D(LOU,LIN)
100     CONTINUE
        A3=AUX(7,1)
        B2=AUX(8,1)
C
C
C
111     FORMAT(1X,8G10.4)
C
C MODIFIED GRAM-SCHMIDT TIME UPDATE
C
        CALL MATSM(AU,0.,8,8)
        CALL SETA(A)
        CALL CALA(A,K,1)
        CALL CALB(A,K,1)
        DO 10 I=1,6
        DO 10 J=1,6
10      AU(I,J)=A(I,J)
        AU(7,7)=1
        AU(8,8)=1
C
        CALL MATMUL(AU,AUX,AUX,8,8,1)
C
        AU(2,7)=XN(1,1)
        AU(4,7)=XN(3,1)+XN(4,1)
        AU(4,8)=-K(3,1)*XN(3,1)-K(4,1)*XN(4,1)
        AU(6,7)=XN(5,1)+XN(6,1)
        AU(6,8)=-K(5,1)*XN(5,1)-K(6,1)*XN(6,1)
        DO 7 KL=1,6
7       AU(2,8)=AU(2,8)-K(KL,1)*XN(KL,1)
C
        CALL MATMUL(AU,U,WP,8,8,8)
        DO 55 JL=1,8
        DO 55 KL=1,8
        WD(JL,KL)=WP(JL,KL)
55      DB(JL,KL)=D(JL,KL)
        DO 60 JJ=8,1,-1
        DO 65 KL=1,9
65      RWD(1,KL)=WD(JJ,KL)
        CALL MATTNS(RWD,TRWD,1,9)
        CALL MATMUL(DB,TRWD,TRWD,9,9,1)
        CALL MATMUL(RWD,TRWD,DT(JJ,JJ),1,9,1)
        IF (DT(JJ,JJ).EQ.0) GO TO 60
C
        DO 80 KK=1,JJ-1
        DO 85 LN=1,9
85      RWD(1,LN)=WD(KK,LN)
        CALL MATMUL(RWD,TRWD,US,1,9,1)
        UT(KK,JJ)=US/DT(JJ,JJ)
```

```
C
C
        DO 95 LK=1,9
95      WD(KK,LK)=WD(KK,LK)-UT(KK,JJ)*WD(JJ,LK)
C
80      CONTINUE
C
60      CONTINUE
        RETURN
        END
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C ***********************************
C
C *******     MAIN PROGRAM     *********
C
C ***********************************
C
C RA3, RB2 ARE PLANT PARAMETER VALUES
C A3, B2 ARE THE ESTIMATES FOR THE PARAMETERS
C
C
        COMMON A1,A2,A3,A4,B2,RA3,RB2
        REAL XN(6,1)
        REAL A(6,6), B(6,1), P(6,6), C(6,6), F(6,6)
        REAL  DV(6,1), E(1,6), K(6,20)
        REAL GA(6,1), AG(1,6), QD(6,6)
        REAL Z(6,1), RA(6,6)
        REAL JO(4), PCO(6,6)
        REAL PE(6,6)
        REAL JTR(4)
        REAL RINV(6,6)
        REAL AUX(8,1), KF(8,6), H(6,8)
        REAL AU(8,8)
        REAL UT(8,8), DT(8,8)
        REAL TIME(100), X1(100), X2(100), U(100)
        REAL TH1(100), TH2(100)
C
C
        MU=5
        TIME(1)=0.
        R=0.5
        AL=1
```

```
C
C PLANT PARAMETER VALUES
      RA3=0.5
      RB2=-0.4
C
C B2 IS INITIAL ESTIMATE FOR RB2
C A3 IS INITIAL ESTIMATE FOR RA3
C
      B2=-0.42
      A1=1
      A2=0.01
      A3=0.485
C
C THE Z VECTOR IS THE INITIAL STATE CONDITION
C AND ALSO THE INITIAL MEASUREMENT FOR
C KALMAN FILTER
C
      A4=1
      Z(1,1)=0.005
      Z(2,1)=0.05
C
C VW IS INITIAL VARIANCE OF STATE FOR FILTER
C
      VW=ABS(Z(2,1)/10.)
      Z(3,1)=0.
      Z(4,1)=0.
      Z(5,1)=0.
      Z(6,1)=0.
      TYPE 111
      TYPE 1
1     FORMAT(2X,'INITIAL CONDITIONS :',/)
      TYPE 111, Z
      TYPE 111
C
C INITIALIZE THE MATRIX H, U-D FACTOR FOR
C INITIAL COVARIANCE FOR THE FILTER
C
      DO 16 I=1,8
      AUX(I,1)=0
      DO 16 J=1,8
      IF (I.LE.6) H(I,J)=0
      IF ((I.LE.6).AND.(I.EQ.J)) H(I,J)=1.
      DT(I,J)=0
      IF (I.EQ.J) UT(I,J)=1.
      IF (I.EQ.J) DT(I,J)=VW
      IF ((I.EQ.J).AND.(I.GE.7)) DT(I,J)=0.1
16    CONTINUE
C
C VNW IS VARIANCE OF NOISE
      VNW=0.01
C
      AUX(7,1)=A3
      AUX(8,1)=B2
C
      DO 2 L=1,6
```

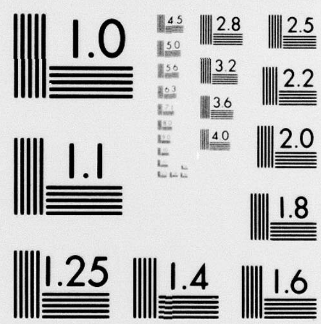
```
2       B(L,1)=0
        B(2,1)=B2
C
C TO GENERATE FILE FOR PLOTTING
C
        TYPE 121
121     FORMAT(' INPUT FILE NUMBER FOR GENERATING GRAPH
     1  IN I2 FORMAT')
C
        ACCEPT 22, NF
22      FORMAT(I2)
C
C
3       TYPE 4
4       FORMAT(' INPUT WNO')
C
C INPUT MAXIMUM WEIGHT FOR MATRIX W
C
        ACCEPT 5, WNO
5       FORMAT(F10.4)
        IF (WNO.GT.1.) STOP
        TYPE 6, WNO
6       FORMAT(/,'    WNO =   ',F10.3,/)
        INW=0
C
C INITIALIZE NOISE MATRIX, STATE WEIGHTING MATRIX,
C NOMINAL TRAJECTORY OF FILTER, AND NOISE
C VARIANCE FOR OBSERVATION: RINV
C
        DO 7 M=1,6
        XN(M,1)=Z(M,1)
        GA(M,1)=0
        IF(M.EQ.2) GA(M,1)=A2
        DO 7 N=1,6
        QD(M,N)=0
        RINV(M,N)=0
        IF (M.EQ.N) RINV(M,N)=VNW/100000.
7       CONTINUE
        QD(1,1)=400
        QD(2,2)=0.2
C
        ICOUNT=1
100     DO 12 KIND=1,4
        KINDT=KIND
        CALL SWQD(QD,WNO,KINDT,AL)
C
C COMPUTATION OF K, THE FEEDBACK GAINS
C
        CALL SETA(A)
        CALL COMKUD(A,B,QD,K,R,MU)
C
C
111     FORMAT(1X,8G10.4)
C
C COMPUTATION OF OPTIMAL COST
```

```
      CALL COMPJ(Z,A,K,QD,GA,VNW,JO(KIND),MU)
12    CONTINUE
      TYPE 13
13    FORMAT(2X,'OPTIMAL COST FOR DIFFERENT WNO
     1 POSITION',/)
      TYPE 111, JO
      TYPE 111
      ISW=4
      SJO=JO(4)
      DO 10 I=3,1,-1
      IF (SJO.LE.JO(I)) GO TO 10
      SJO=JO(I)
      ISW=I
10    CONTINUE
      TYPE 111, ISW
      IF (ISW.EQ.4) GO TO 11
      CALL SWQD(QD,WNO,ISW,AL)
      CALL SETA(A)
      CALL COMKUD(A,B,QD,K,R,MU)
C
C
C
C TO GENERATE DATA FOR PLOTTING
C
11    X1(ICOUNT)=Z(1,1)
      X2(ICOUNT)=Z(2,1)
      TH1(ICOUNT)=A3
      TH2(ICOUNT)=B2
      DRA3=RA3-A3
      DRB2=RB2-B2
      WRITE(NF,202) TIME(ICOUNT),X1(ICOUNT),X2(ICOUNT)
      NF1=NF+1
      WRITE(NF1,202) TIME(ICOUNT),DRA3,TH1(ICOUNT),
     1 DRB2, TH2(ICOUNT)
202   FORMAT(5(1X,G12.4))
C
C
C
C KALMAN FILTER TO OBTAIN ESTIMATES A3, B2
C
      CALL KALMAN(KF,AUX,UT,DT,Z,K,RINV,H,VNW,AU,XN)
      CALL WDSK(AU,8,8,4)
      TYPE 111, AUX
      TYPE 111, A3, B2, ICOUNT
      CALL SETA(A)
      CALL CALA(A,K,1)
      CALL CALB(A,K,1)
      CALL MATMUL(A,XN,XN,6,6,1)
C
C COMPUTE NEXT STATE USING PLANT EQUATION
C
C
      CALL SETRA(RA)
      CALL CALRA(RA,K,1)
      CALL CALRB(RA,K,1)
```

AD-A077 148 ILLINOIS UNIV AT URBANA-CHAMPAIGN DECISION AND CONTROL LAB F/G 13/8
SENSITIVITY ADAPTIVE CONTROL OF A MAGNETIC SUSPENSION SYSTEM.(U)
MAY 79 Y M CHAN DAAG29-78-C-0016
UNCLASSIFIED DC-25 NL

2 OF 2
AD
A077148

END
DATE
FILMED
3-81
DTIC



MICROCOPY RESOLUTION TEST CHART

```
C
C
C WRITE A MATRIX ON DISK FOR LAS ANALYSIS
        CALL WDSK(RA,6,6,3)
C
C
        TYPE 111, (K(JJ,1),JJ=1,6)
        DO 15 JK=1,6
15      E(1,JK)=K(JK,1)
        CALL MATMUL(E,Z,U(ICOUNT),1,6,1)
        TYPE 111, U(ICOUNT)
C
C GENERATE DATA FOR PLOTTING
        NF2=NF+2
        WRITE(NF2,202) TIME(ICOUNT), U(ICOUNT)
C
        CALL MATMUL(RA,Z,Z,6,6,1)
        INW=INW+1
        CALL RANDOM(INW,VNW,0.,W,KKK)
        TYPE 111, W
        DO 17 JL=1,6
17      DV(JL,1)=GA(JL,1)*W
        CALL MATAS(Z,DV,Z,6,1,1)
        TYPE 111, Z
C
C
C
        ACOUNT=ICOUNT
C
C THE FOLLOWING STATEMENTS WITH COMMENT ON
C FIRST COLUMN ARE FOR CHANGING PARAMETER
C AT T=0.2 AND PERTURBATION OF STATES AT
C THAT INSTANCE
C
C
C       IF (ICOUNT.EQ.20) RA3=0.4
C       IF (ICOUNT.EQ.20) RB2=-0.3
C       IF (ICOUNT.EQ.20) Z(1,1)=Z(1,1)+0.005
C       IF (ICOUNT.EQ.20) Z(2,1)=Z(2,1)+0.01
C
        IF (ICOUNT.EQ.41) STOP
        TIME(ICOUNT + 1)= A2 + TIME(ICOUNT)
        ICOUNT=ICOUNT+1
        GO TO 100
        END
```

DATE

FILMED

-8